E-3708U
FINAL Cycle 1 Functional Requirements Specification

Functional Area VI - Army-Wide Assessment of MPT Issues

Task Order #09
Development of the Conceptual Model of the Mission Space (Cycle 1 CMMS) Prototype Implementation

under the
MANPRINT Support Services (MSS) Contract

Contract Number:
DAAL01-95-C-0115

Date of Preparation: 21 June, 1996

Prepared By:

Dynamics Research Corporation
60 Frontage Road
Andover, MA 01810

Prepared For:

U.S. Army Research Laboratory (ARL)
Aberdeen Proving Ground, MD 21005-5425

Sponsored by:

Defense Modeling and Simulation Office (DMSO)
Alexandria, VA 22311

UNCLASSIFIED

Approvals

Dr. Lawrence H. O'Brien, **DRC Principal Investigator**	**Date**
Dr. Michael Wagner **DRC Program Manager**	**Date**
Dr. David Promisel **ARL COR**	**Date**
Mr. Jack Sheehan **DMSO Cycle 1 CMMS Technical Director**	**Date**
Mark Jefferson, Lt. Col., USAF **DMSO Cycle 1 CMMS Program Manager**	**Date**

SECTION 1-INTRODUCTION

1.1 Objective and Document Organization

This document describes the functional requirements specification for the Cycle 1 prototype of the Conceptual Model of the Mission Space (CMMS).

The specification will be used by members of the Cycle 1 CMMS development team including DRC, its subcontractors, and the Government technical and management personnel. DMSO will distribute this specification to selected programs and projects, both DMSO managed and externally sponsored, as a template for related activities.

1.2 Referenced Documents

The following documents provided the primary sources for Cycle 1 requirements.

Document Name	Revision	Date
Cycle 1 CMMS Prototype Development Plan	0	30 March 96
Conceptual Model of the Mission Space Technical Framework	6.2	20 May 96

Table 1 Referenced Documentation

The Technical Framework, which is included as Attachment A, describes the overall CMMS program and the CMMS prototype effort and defines key terms used in this document. The Development Plan describes the development process that will be used during the prototype and the role of the functional specification in that process. An abbreviated version of the development plan is included in Appendix B.

1.3 Cycle 1 Operational Concept

The objective of the Cycle 1 effort is to demonstrate all major CMMS functions using a realistic mission thread. Functions will be demonstrated at DMSO and then at selected major simulation program sites. Key activities both prior to and during the demonstration are listed below.

Prior To Demonstration. The *Data Manager*s (DRC and S3I) will work with 4 major simulation programs to ensure that proper knowledge is created to support the demonstration. The four major programs are: JSIMS, WARSIM, NASM, and Navy JWARS. The *Data Manager*s will outline a thread for a Joint Interdiction mission. They will then contact the simulation programs and determine if the mission space data is available to support the thread. If resources and schedule permit, the simulation program will be asked to fill in missing data. If the simulation

program cannot provide the missing data, the *Data Manager*s will work with the simulation program to provide the missing data.
The *Data Manager*s will also work with the 4 major simulation programs to develop a DIF for each program.[1] This will involve: (1) getting the simulation program to export the thread data to a readable file format, (2) capturing information on the exported data structure and scheme, (3) where possible, obtaining data dictionary link files which link the program's data dictionary to CMMS data dictionary files, (4) where required, adding fields to selected program files to link to CMMS data dictionary items, and (5) performing additional refinements to support the DIF.

<u>During Demonstration.</u> All nine major CMMS functions *(Create, Convert, Integrate, Locate, Register, Extract, Release, Evaluate, Manage*) with the exception of *Create* will be demonstrated using the Joint Interdiction thread.[2] The *Create* function while part of the overall CMMS concept is not part of the CMMS prototype software. Per the Technical Framework, individual simulation programs are expected to create knowledge using knowledge capturing software appropriate for their own individual project objectives. The knowledge created by these tools will provide the input to the CMMS prototype software. From a user interface perspective, *Convert* and *Integrate* will be initiated by a single dialog. However, the functions will be implemented by separate algorithms. During the demonstration inputs and outputs to each algorithm will be reviewed.

More specifically, the *Data Manager*s will first present a briefing describing the thread and the inputs from the simulation programs. Then the *Data Manager*s will walk through each CMMS function using a pre-defined script. The *Data Manager*s will role play the contributors and simulation developers. Listed below is an outline of the items that will be included in the demonstration. A more detailed description of these items is provided in Section 2.

Locate (View)
-Main CMMS Dialog(s)
Register
-Model Description Dialog
-Model User Description Dialog
-Model Loading and Data Checking Dialog
-Model Update Dialog(s)
Convert & Integrate
-Convert and Integrate Dialog (which initiates the Convert and Integrate algorithms)
-Model Semantic Assignment Report (shows results of integrate function)
Release
-User Sign-On Dialog
Locate (View)
-Main CMMS Dialog(s)

---

[1]The Modeling and Simulation Functional Data Administrator (MS Fdad) will eventually provide guidance for the generic content of DIFs but the CMMS Data Managers must apply this guidance to develop specific DIFs for particular simulation programs. For Cycle 1, not even the generic guidance is expected to be available.
[2]A detailed definition of each function is provided in CMMS Technical Framework which is listed in Appendix A.

-Information Interaction Fully Structured View
--(Used to show Joint Interdiction Thread and results of convert and integrate)
Extract
-Query: Model Selection Query (helps users find a relevant model)
-Reports: Organization Mission Task and Task Data Reports
-Download Report Data Dialog
Evaluate
-Update VV&A Pedigree Dialog
--(Information Interaction FSV to show coding to indicate change in VV&A Status)
Manage
-Reference Document Update Dialog

During demonstration, user comments will be recorded.

After Demonstration Functional and architectural specifications will be updated to reflect user comments.

SECTION 2.0 REQUIREMENTS

This section is divided into 9 subsections corresponding to the nine major CMMS functions identified in the CMMS Technical Framework. [3] Figure 1 below describes the relationships between the 9 major CMMS functions (Create, Register, Convert, Integrate, Locate, Extract, Release, Manage, and Evaluate), the 4 major data management roles (Producer, Consumer, Data Manager, and Examiner), and the two major data elements (raw simulation model and common CMMS data base).

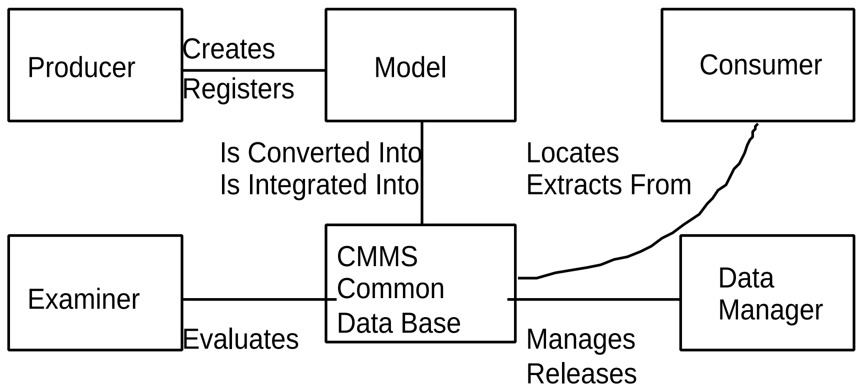


Figure 1 Cycle 1 Context Diagram

2.1 Create

As in the operational CMMS, the Cycle 1 prototype will not provide automated tools for capturing or creating knowledge. It is assumed that individual model *Producer*s will create mission space knowledge using COTS tools selected to meet individual simulation program needs. During Cycle 1, the goal shall be to create no new mission space knowledge but to utilize knowledge obtained from existing simulation programs.

2.1.1 Cycle 1 Mission Space Thread

Capabilities for the Cycle 1 prototype shall be demonstrated using a realistic mission space "thread." The thread shall represent a Joint Interdiction operations and shall include Air Interdiction, Land Interdiction, and Maritime Interdiction missions for the air, land, and maritime components, respectively.

---

[3] A detailed definition of each function is provided in CMMS Technical Framework which is listed in Appendix A.

The knowledge needed to populate the Joint Interdiction thread shall be obtained from the following simulation programs:

Operation\Mission	**Program**
Joint Interdiction (down to component interdiction mission)	JSIMS
Air Interdiction (down to squadron level)	NASM
Land Interdiction (down to platform level)	WARSIM
Maritime Interdiction (down to ship level)	JWARS

2.1.1.1 Description of Joint Interdiction Scenario

Cycle 1 will use a Joint Interdiction operation involving a Major Regional Conflict (MRC). The scenario will be based in Korea in the year 1998. A more detailed description of the scenario is provided in Appendix C.

Joint Publication 3-03, Doctrine for Joint Interdiction Operations (currently pending final approval), other joint publications listed in Appendix A, References, of Joint Publication 3-03, and applicable Service doctrine and Tactics, Techniques, and Procedures will be used to detail activities at the Strategic, Operational, and Tactical levels of war to outline the thread. Joint tasks identified in the Universal Joint Task List (CJCSM 3500.04, 15 May 1995) and other tasks reflected in appropriate Mission Training Plans will be used as higher level tasks.

Interdiction involves one or more of the following: *diverting* enemy forces intended for battle areas where they are immediately or critically required, *disrupting* the movement of and routing of the enemy's information, materiel, forces and supplies, *delaying* enemy forces such as when they bunch up behind a damaged route segment or are forced to make lengthy detours, or *destroying* the enemy's vital resources and infrastructure by lethal or non-lethal means. All four of these interdiction objectives will be represented in the scenario and will be achieved through the concept of Joint Precision Interdiction and the employment of interdiction targeting considerations. Targets and interdiction assets will be matched in deliberate planning through the Joint Force Air Component Commander and the Joint Target Coordination Board. Targets will be identified in the scenario that will lend themselves to attack by land, aerospace, or maritime assets. At that time "hooks" will be provided into simulation programs identified above to allow for both greater resolution and refinement. The intent is to take portions of the interdiction "thread" to the warfighter level or individual platform level.

During the demonstration, the objectives will be to: (1) *register, convert,* and *integrate* individual simulation program data related to the thread, and (2) demonstrate the remaining functions (*locate*, *extract*, *release*, *manage*, and *evaluate*) using integrated thread data. The information interaction *Fully Structured View* will be the key to demonstrating the latter. The information interaction FSV will allow users to decompose the Joint Interdiction Mission down to the Land, Air, and Maritime interdiction missions, and then decompose selected tasks within each of these missions down to the level of an individual platform's tasks. The information interaction FSV will also be capable of showing task inputs and outputs.

2.1.2 Content of Cycle 1 CMMS Data Base

2.1.2.1 Mission Space Data To Be Obtained from Simulation Programs

The focus of Cycle 1 CMMS shall be on the storage and distribution of information on key organizations in the mission space, their associated missions and tasks, and the informational inputs and outputs between tasks.

The Cycle 1 CMMS shall contain information on generic types of organizations (e.g., Armor battalion) rather than specific organizations (101st Armor battalion).

The Cycle 1 CMMS shall describe tasks and missions for entities at various levels of abstraction ranging from a Joint Task force down to individual platforms or weapon systems. However, in accordance with the "thread" concept, only a subset of missions and tasks for selected organizations will be broken down to the platform level.

The CMMS Technical Framework defines five levels of war. The Cycle 1 CMMS shall include tasks at the three CMMS levels of war corresponding to the UTJL strategic, operational, and tactical levels of war as well as individual service tasks down to the platform level.

Cycle 1 CMMS shall only contain data on US Military Forces. Allied, opposing, and civilian forces shall *not* be represented in Cycle 1 CMMS.

The following information shall be obtained from each simulation program for each mission/operation thread.

- List of major organizations involved in the mission and their key attributes including organization id and organization name.
- Breakout or decomposition of organizations into subordinate organizations and their key attributes
- List of tasks associated with each operation or mission and their key attributes including task id, task name, performing organization(s), organizational responsibilities for task, preconditions or entrance criteria (including triggers or entrance criteria), postconditions (including exit criteria), resources, modified objects, timing factors, references, rationale/description, and conditions impacting task performance.
- Breakout or decomposition of tasks into lower level tasks

- Inputs and outputs between tasks including input/output name, sending/receiving organizations, and reference

2.1.2.2 Management Data

In addition to storing mission space data, Cycle 1 CMMS shall store additional data elements to assist in the management of this data including the following data elements which shall be stored at the model level

- Persons associated with a model to include name, address, email, phone numbers, and assigned management roles.
- Model data source (production pedigree, model producer sponsor, and model producer)
- Model authoritative data source (data source plus VV&A Pedigree level, VV&A sponsor, and VV&A Examiner).

Cycle 1 CMMS shall include the following management libraries:

- VV&A Pedigree Level (Not Verified, Verified, Validated, Accredited, and Questionable due to change in associated reference document)
- Clearance Levels (Unclassified, Confidential, Secret, Top Secret)
- Management Roles (*Producer*, *Sponsor*, *Examiner*, *Consumer*, *Data Manager*)[4]

2.2 Register

2.2.1 Model Registration/Update Descriptions

[4]Definitions of the management roles is provided in the CMMS Technical Framework listed in Appendix A.

Cycle 1 CMMS shall provide *Producers*, who have been granted model registration/update privileges, the capability to register a new model or register an update to an existing model.

When *Producers* register they will either select an existing model for update or indicate that they want to create a new model. If *Producers* indicate that they want to create a new model, they shall then indicate which of the existing Data Interchange Formats (DIFs), if any, that model will use.

While adding a new model to Cycle 1 CMMS, the *Producer* shall provide summary information on that model to include the following: model name, version number, version description, and production pedigree method (i.e., textual description of the method used to produce the model) model producer sponsoring organization, and model producer organization, VV&A sponsor organization, and VV&A examiner organization.

If a *Producer* indicates that the new model will not use any of the existing DIFs, he/she will be provided with dialog describing how to contact the Cycle 1 CMMS *Data Managers.*

Producers updating existing models shall provide a description of the update to include the following: version number, version name, and updates to other model descriptive data.

Model updates shall only be required when there are substantive changes or additions to the mission model space. Changes which simply involve updates to verification and validation data by an *examiner* shall not require registration of a model update.

2.2.2 Model User Descriptions

As part of the Cycle 1 registration process, the *producer* shall describe other persons who will have access to the model data and define the type of access they will be provided. Personnel involved in all management roles will be described in terms of the following:

- Personal Data
  -NAME: First Name, Last Name, Middle Initial
  -ADDRESS (Organization, Street, City, Zip Code)
  -PHONE NUMBER
  -FAX NUMBER
  -EMAIL ADDRESS
  -ORGANIZATION
- Management Role(s) (*Producer*, *Sponsor*, *Examiner*, *Consumer*, or *Data Manager*)
- Clearance Level

2.2.3 Loading & Storing Model

Once a model description has been completed, the Cycle 1 CMMS shall allow users to load the model by specifying the location of the files on the CMMS server. The Cycle 1 CMMS software shall then store the files within CMMS.

2.2.4 Data Quality Checking

After a model has been loaded, the quality of the model data shall be examined by comparing it against data structure information contained in the DIF. This shall involve: (1) determining if appropriate data is stored in a field given the assigned data type, and (2) determining if data in a field falls within specified ranges for that field.

2.3 Convert

Cycle 1 CMMS shall provide automated capabilities for converting data from individual *models* into the common mission space model data base using a Data Interchange Formats (DIF).

2.3.1 Data Interchange Formats

Cycle 1 CMMS shall store data on data interchange formats (DIFs).

The Cycle 1 DIF shall include:

(1) information or attributes about the data base as an aggregated whole (e.g., date when data base version was released, name of data base or files, version description,),

(2) information about the actual data elements stored in the data base such as its schema, data structure and data element names, types, measurement units and range values.

(3) mappings or linkages between the data elements (e.g., file, record, field) in the simulation data base and data elements (e.g., file, record, field) in the common mission space data base.

(4) methods for converting data elements in the two data bases which vary in terms of data type or size and legal ranges for specified data elements.

During Cycle 1, the following DIFs shall be developed and applied.

Program	**Source Format**
JSIMS	Rational Rose
NASM	RDD 100
WARSIM	UNISQL
NAVY JWARS	STATEMATE

2.4 Integrate

The objective of the *Integrate* function is place model data into a common data base structure and semantic framework. [5]

The Cycle 1 CMMS shall contain a data dictionary. The Cycle 1 single-thread data dictionary shall consist of a set of library files on the following elements:

<u>Entity Library</u>
-Organizations
-Inputs and Outputs
<u>Action Libraries</u>
-Actions
-UJTL Tasks
-Operations

The Cycle 1 CMMS shall read and parse task statements from model input file(s) and compare them to information in the data dictionary to identify equivalent Actions.[6]

The Cycle 1 CMMS shall read and parse the highest level tasks in each organization's mission from the model input file(s) and compare to them to information in the data dictionary to identify equivalent UJTL tasks.

The Cycle 1 CMMS shall read and parse performing organizations from the model input files and compare them to information in the data dictionary to identify equivalent Organizations.

The Cycle 1 CMMS shall read and parse inputs and outputs from the model input files and compare them to information in the data dictionary to identify equivalent Inputs and Outputs.

The Cycle 1 CMMS shall provide *Producers* with an on-line interface which lists items from the original data file (e.g., tasks and organizations) and the assignment of equivalent items from the data dictionary libraries. This interface shall also allow users to modify existing library assignments or make new assignments where no assignments have been identified. Selected users shall also have the option of adding model items with no current library assignments to the CMMS data dictionary.

2.4.1 Relationship Checking

[5]A more formal definition of the *Integrate* function is provided in the Technical Framework listed in Appendix A.
[6]To successfully perform the *Integrate* function in a timely manner, agreement on the items to included in the data dictionary library must be achieved by July 1.

After the data from a model has been integrated into the common data base, the Cycle 1 CMMS shall check the completeness of the model's data relationships. This shall involve identifying data elements which lack required links or relationships to other data elements (e.g., task input or output does not have both a sending and receiving organization).

2.5 Manage

2.5.1 Configuration Management/Version Control

All data in the unified CMMS data base shall be traceable back to a producer model and version.

The Cycle 1 unified CMMS data base will store complete data sets for each model and version.

Model and version shall be identified in on-line reports and queries where appropriate.

2.5.2 Traceability of Changes to Reference Documents

The Cycle 1 CMMS mission space data base shall list reference documents for each Joint mission or operation, individual tasks, and task inputs and outputs.

The Cycle 1 CMMS shall allow *Data Managers* to indicate that a reference document has been updated.

When a reference document is updated, Cycle 1 CMMS shall automatically modify the VV&A pedigree of data elements associated with the updated reference document to indicate that they are potentially invalid.

Producers and *examiners* may then evaluate whether or not the reference updates do in fact impact the validity of the data element and then use the on-line VV&A update capability to change the VV&A pedigree.

2.5.3 Data Storage

The Cycle 1 CMMS shall be implemented using a COTS relational DBMS.

Ideally for Cycle 1, the goal will be to store Cycle 1 CMMS mission space data in a manner consistent with the canonical form described in the *Cycle 1 CMMS Technical Framework*. The canonical form provides a generic logical data structure for storing mission space data. Cycle 1 management data is *not* required to be stored in the canonical form.

2.5.4 Event Services

Cycle 1 CMMS shall provide a Pull Event Service to create Integration reports for Model Consumers.

Cycle 1 CMMS shall provide a Pull Event Service to create Release reports for Model Producers.

Cycle 1 CMMS shall provide a Push Event Service to notify Model Producers that an Integrated Model has been created using a Model Registered by that Producer.

Cycle 1 CMMS shall provide a Push Event Service to notify Model Consumers that a previously Released Model is potentially invalid due to the Registration of a Model revision by a Producer or re-Integration of a Model by a Manager.

2.5.5 Installation Instructions

As part of Cycle 1 effort, the Data Manager (DRC) will develop instructions for installing the CMMS data base and software. Where appropriate, data base installation instructions will be congruent with concepts and guidelines described in the *GCCS Central Data Server Development and Integration Standards, January 31, 1996.*

2.6 Release

2.6.1 User/Identification Access Control

Producers and *simulation developers* shall have direct access to the Cycle 1 CMMS through the INTERNET. [7]

Cycle 1 CMMS *Data Managers* shall have a direct local interface to Cycle 1 CMMS.

The Cycle 1 CMMS shall provide capabilities for registering and authenticating Cycle 1 CMMS users. During Cycle 1, users shall only be able to be registered as part of the model registration process.

The Cycle 1 CMMS shall only allow users to only view data on those models for which they have been authorized and shall only allow them to perform actions on those models which are consistent with their assigned management roles.

2.6.2 Security Services

The Cycle 1 CMMS shall be implemented using a operating system and associated file system that supports C2 security level.

[7] During Cycle 2, CMMS shall be accessible through a MSRR Web page.

2.7 Locate (View)

The Cycle 1 CMMS shall allow users to locate and view data in the integrated data base via fully structured views or tabular reports.

2.7.1 Fully Structured Views

A "Fully Structured View" or FSV provides a graphical user interface for viewing one aspect of mission space knowledge. [8]

Cycle 1 CMMS shall provide one fully-structured view (FSV) to allow *simulation developers* to view and assess the information in the common Cycle 1 CMMS data base. The information interaction FSV shall be displayed during Cycle 1.

The FSV shall be read-only.

Graphical displays accessible via a "point and click" interface shall be the central organizing theme for displaying information in all FSVs.

2.7.1.1 Information Interaction FSV

The Information Interaction FSV (IIFSV) shall describe the informational inputs and outputs among organizations and among the tasks within an organizations.

The IIFSV shall:

- Allow users to select an organization
- View the tasks associated with the selected organization or a specific mission of the organization
- View the inputs/outputs to the organization and to the each of the organization's tasks
- Traverse and view the flow of information between tasks or organizations by clicking on elements of the graphical interface.

2.7.2 On-line Reports

CMMS shall provide a capability for users to view and print on-line reports on mission space and management data.

Reports shall have filters which allow users to select the items to be included in the report.

The following mission space data reports shall be supported during Cycle 1:

[8] A more formal definition of a *Fully Structured View* is provided in the Technical Framework listed in Appendix A.

Organization Mission Tasks--List of tasks for an organization and mission selected by the user.

Task Data--Detailed listing of task attributes for an organization, mission, and task(s) selected by the user.

The following management data reports shall be supported during Cycle 1:

Model Semantic Assignments-Listing of key model terms (either Actions, Organizations, or Inputs/Outputs) and the equivalent item from the CMMS data dictionary.

Model Data Quality Errors--Lists fields from files and records in the simulation model input files which have data quality errors (missing data, wrong data type in field, or value out of range).

Model Relationship Errors. Lists model element from integrated data base and missing relationship.

2.8 Extract

2.8.1 On-line Queries

Cycle 1 CMMS shall provide a capability for simulation developers to perform on-line forms-based queries. The following queries shall be supported during Cycle 1.

- Model Selection--Query to find models involving particular organizations, missions, inputs/outputs, and/or UJTL tasks.

Cycle 1 CMMS shall provide a capability for users to view the results of the query on-line.

2.8.2 Downloading of Data

Cycle 1 CMMS shall provide a capability for users to download the data in a report or a query to the user's local processor.

2.9 Evaluate

2.9.1 Evaluate and Update VV&A Pedigree

Cycle 1 CMMS shall provide an on-line interface to allow *examiners* to update *the VV& A Pedigree* of individual organizations and their associated tasks and missions in the common mission space data base.

ATTACHMENT A: CYCLE 1 CMMS TECHNICAL FRAMEWORK

Defense Modeling and Simulation Office

1901 N. Beauregard St., Suite 504, Alexandria, VA 22311

Conceptual Models of the Mission Space (CMMS)

Technical Framework

Document number:	ARL/UT-ITG-DMSO-0002
Revision number:	0.1.6
Date:	24 APR 1996

Prepared by:

Furman Haddix ARL:UT	signature	date
Robert Might George Mason University	signature	date
Jack Sheehan ARL:UT	signature	date
Ronald Weller SAIC Alexandria	signature	date

Approvals:

Lt. Col. Mark Jefferson CMMS Program Manager	signature	date
Jack Sheehan CMMS Technical Director	signature	date

Revision History

Revision	Date	Author	Description
0.1.1	13 MAR 1996	Jack Sheehan, et al	initial draft
0.1.2	04 APR 1996	Jack Sheehan, et al	re-focus on CMMS content
0.1.3	12 APR 1996	Jack Sheehan, et al	define common syntax and semantics
0.1.4	19 APR 1996	Jack Sheehan, et al	resolve internal DMSO comments
0.1.5	24 APR 1996	Jack Sheehan, et al	re-arrange sections to change order of presentation to begin with most concrete and end with more abstract concepts, modify interactions, replace transaction with task
0.1.6	25 APR 1996	Jack Sheehan, et al	change direct-object, task, and interaction definitions

1. PURPOSE

The Defense Modeling and Simulation Office (DMSO) is leading an effort to develop Conceptual Models of the Mission Space (CMMS) as directed in the DoD Modeling & Simulation Master Plan [1]. CMMS is a simulation-independent first abstraction of the real world of activities associated with a particular set of missions. The focus of the DMSO CMMS project will be on the military operation mission space. CMMS will provide rigorous specifications of military operations for the simulation developer which are

- derived from authoritative sources,
- described using common syntax and semantics, and
- independent of any particular simulation implementation.

In particular, CMMS will

- provide a Data Interchange Format (DIF) [8] for the conversion, integration, storage, and extraction of these models of the mission space;
- support a variety of structured views for the display and manipulation of models;
- employ the Modeling and Simulation Resource Repository (MSRR) [9] to provide physical access and logical connectivity to models, and
- specify an entities, actions, tasks, and interactions (EATI) representation of the military mission space which is independent of any specific CASE tool or utility employed to capture it.

This paper presents the CMMS Technical Framework, Version 0.1. The CMMS Technical Framework specifies the definitions, content, structure, process, and infrastructure required for the creation, management, and distribution of conceptual mission models for use by DoD simulation developers.

CMMS is a work-in-progress by DMSO in conjunction with the major DoD Modeling & Simulation (M&S) development programs, particularly the Joint Simulation System (JSIMS) and the Joint Warfare System (JWARS). In addition to JSIMS and JWARS, DMSO anticipates that a number of other joint and Service M&S programs will make significant contributions to CMMS including: Air Force NASM, Army WarSim, ARPA STOW-97, Navy BFTT and NSS, and Marine Corps Emerald Light. The Defense Intelligence Agency will also provide appropriate contributions to CMMS.

The DMSO sponsored CMMS Technical Working Group (TWG) has provided an informal process for developing a knowledge acquisition consensus among the DoD M&S programs noted above. The CMMS Technical Framework presented here provides a formal mechanism for coordination and cooperation in knowledge acquisition and maintenance to support simulation developers. As a work-in-progress, Version 0.1 of the Technical Framework specifies requirements which are considered to be necessary for mission space conceptual model compatibility and re-use among the major DoD M&S programs. DMSO has initiated the CMMS Prototype project to provide direct feedback on the sufficiency of the requirements stated here.

Section 2 provides the Technical Framework requirements. Specifications captioned <u>Minimum Requirements</u> are considered mandatory for registration and eventual inclusion in CMMS. Specifications marked <u>Requirement Extensions</u> are considered preferred practices which will substantially enhance CMMS integration and re-use. As further experience is gained in CMMS, it is expected that many of the preferred practices identified under <u>Requirement Extensions</u> will migrate to mandatory compliance under <u>Minimum Requirements</u>. References are provided in Section 3. The Appendix provides an entity-relationship data model using IDEF1X notation describing the EATI representation.

2. TECHNICAL FRAMEWORK

Reserve Word A specific term or concept which is defined and used to specify the CMMS Technical Framework. These terms will be italicized and the first letter capitalized when used as a *Reserve Word.*

Following the DoD M&S Master Plan [1], the definitions provided in the Glossary of M&S Terms [2] and the DoD Data Dictionary System [3] are included here as *Reserve Words* by reference. Changes or extensions to these external definitions in this CMMS Technical Framework will be explicitly noted in the definitions provided here.

Model A physical, mathematical, or otherwise logical representation of a system, entity, phenomenon, or process [2].

Mission A *Task* which is executed by an *Actor* to achieve a specific *Objective*. A *Mission* includes the specific entrance criteria and exit criteria which govern its initiation and termination as well as the specific measures of performance and effectiveness which indicate its relative success. *Objective* is defined [5] and [7]. *Task* and *Actor* are defined in Section 2.5 below.

Mission Space A set of *Missions* which share a common organizing principle, purpose, or feature.

The CMMS Technical Framework specifies the definitions, content, structure, process, and infrastructure required for the creation, management, and distribution of the Conceptual *Models* of the *Mission* Space (CMMS) for use by DoD simulation developers. This Technical Framework specifically addresses the military operations *Mission Space*. Many of the principles established here are equally valid in other well-defined DoD *Mission Spaces*, such as medical care or manufacturing; however, detailed discussion of these other *Mission Spaces* are beyond the scope of the CMMS Technical Framework Version 0.1.

CMMS captures conceptual *Models* or logical representations but does not provide actual implementations or simulations. Note that *Models* is plural. A *Model* is a logical or conceptual representation of an entity (or phenomenon, or process, etc.) but is not the actual physical entity itself. In developing a conceptual representation, the *Model* developer must select some details of the actual entity for inclusion in the *Model* and must (either explicitly or implicitly) select other entity details for exclusion from the *Model*. In CMMS, it is the *Mission* which drives and determines the inclusion/exclusion choice in *Model* development. Therefore, distinct *Mission Spaces* lead to distinct conceptual *Models* for the same entity (or process, or phenomenon,...) because different sets of Missions will lead to different *Modeling* inclusion/exclusion choices . Within a particular *Mission Space,* there are many *Missions* -- where again -- distinct *Missions* lead to distinct *Models.* Finally, representation of the same specific *Mission* across the real world distinctions in level-of-abstraction (and corresponding level-of-detail) leads to a family of related conceptual *Models* rather than a single *Model*.

The remainder of Section 2 is organized as follows. The CMMS process is summarized in Section 2.1. Section 2.2 defines an *Authoritative Data Source*. Section 2.3 provides the associated definitions of *Authorized Data Consumer*. Section 2.4 introduces the notions of common syntax and semantics in CMMS. Section 2.5 specifies the EATI representation. Section 2.6 extends the EATI representation by defining the levels of warfare used to specify hierarchical context and relationship between entities.

2.1 CMMS Process

Create Capture of real world information about one or more *Missions* in the form of a *Model* for eventual inclusion in CMMS.

Register — Submission of one or more *Models* for actual inclusion in CMMS including source, format, and content checking with deficiency correction as appropriate.

Convert — Transformation of a *Model* which has been *Registered* from its native form to a standard form required by CMMS including extraction of semantic and syntactic elements.

Integrate — The act of combining, normalizing, storing, indexing, and in general migrating *Registered Models* in CMMS standard form to a higher level of structural maturity and semantic enforcement within a unified database.

Manage — The provision of configuration management, version control, change traceability, data storage, and resource allocation for *Models* in CMMS.

Release — The provision of security services, access control, user identification for use and examination of *Models*.

Locate — The use of on-line browsing tools, automated searches, and retrieval queries to identify *Models* of interest.

Extract — The use of application programming interfaces and automated data retrieval services to gather, format, package, and delivery CMMS *Models* to simulation developers.

Evaluate — Determination of the suitability of a CMMS *Model* for a specific end-use by a simulation developer.

There are three primary types of CMMS end-users:

- contributors who *Create* and *Register Models*,
- CMMS data managers who *Convert*, *Integrate*, *Manage*, and *Release Models*, and
- simulation developers who *Locate*, *Extract*, and *Evaluate* CMMS *Models*

The *Create* and *Evaluate* processes are external to CMMS. *Register* and *Extract* are *Model* transfer boundaries to and from CMMS. *Convert* and *Locate* are internal CMMS processes with external GUI interfaces for contributors and simulation developer visibility. *Integrate*, *Manage*, and *Release* are internal processes to CMMS which are not to visible or accessible by external end-users. In addition to the three primary types of CMMS end-users indicated above, DMSO anticipates that a number of secondary end-users will also employ CMMS to conduct:

- VV&A by designated agents,
- military doctrine development by appropriate commands,
- scenario development by simulation end-users, and
- operational planning by warfighters.

DMSO has initiated the CMMS Prototype project to identify detailed technical requirements for this process. This Prototype will implement a single, end-to-end thread through the CMMS process. During the Prototype project,

- Selected knowledge acquisition specialist from the major M&S development programs will provide previously *Created Models* for *Registration.*
- CMMS examples for each required capability will be implemented by developers (DRC, S3I, and IMC) under contract to DMSO.

- Selected simulation developer from the major M&S programs will identify requirements for and provide evaluation of CMMS Prototype functionality.

Minimum Requirements:

Each Model presented for CMMS *Registration* shall comply with the appropriate Modeling and Simulation Resource Repository (MSRR) registration standards.

CMMS shall be capable of *Locating* and *Extracting* each *Model Integrated* into CMMS in compliance with appropriate MSRR standards.

Requirement Extensions:

CMMS shall be capable of *Converting* electronically readable *Models* provided in open standard file formats associated with common word processing applications, Computer Aided Software Development (CASE) tools, and database management systems including but not limited to: MS Rich Text Format, FIPS IDEF0 and IDEF1X formats, industry standard CDIF and STEP formats, and ANSI/ISO SQL-2 and ODBMG-93 formats.

CMMS shall be capable of *Converting, Managing, Releasing, Locating,* and *Extracting* electronically readable data sets which comply with appropriate Data Interchange Format (DIF) standards.

2.2 Authoritative Data Sources

This section describes the CMMS requirements which ensure that the accuracy and authenticity of any particular *Model* is specified in sufficient detail for a simulation developer to ascertain the suitability of that *Model* for that simulation developer's specific requirements.

Sponsor — The combination of a *Person*, *Organization*, and *Role* which constitute the *Actor* which has been assigned the command responsibility for specific content, structure, or process which are required to create, manage, or release CMMS information. *Person*, *Organization*, and *Role* are defined in [3]. *Actor* is defined in Section 2.5 below.

Producer — The combination of a *Person*, *Organization*, and *Role* which constitute the *Actor* who, because of either mission or subject matter expertise, actually creates, manufactures, or constructs specific content, structure, or process for incorporation in CMMS.

Production Pedigree — The comprehensive audit trail which describes the specific methods and procedures actually employed by the Producer to create, derive, and construct a particular *Data* item or *Model* for specified end-use. This *Pedigree* provides *Data Source* traceability for constituent *Data* items and *Models* which were incorporated into or employed to produce the particular *Data* item or *Model* in question.

Data Source (DS) — The combination of *Sponsor*, *Producer*, *Data*, and *Production Pedigree* which provide a *Data* item or *Model*. The *Producer* creates that actual *Data* item or *Model* instance by direction of the *Sponsor* and records these activities in the *Pedigree*. This definition of *Data Source* is a compatible extension of the definitions in [2] and [4].

Examiner	The combination of a *Person*, *Organization*, and *Role* which constitute the *Actor* that actually inspects, tests, and evaluates specific CMMS content, structure, or process for the purpose of verification, validation, certification, or accreditation.
VV&A Pedigree	The comprehensive audit trail which records the formal verification, validation, and accreditation activities actually performed on a particular *Data Source* by the *Examiner*. This *Pedigree* also provides traceability for input *Data* items or *Models* which a.) were employed to produce the actual *Data* items or *Models* provided in the *Data Source* in question but which b.) were not delivered along with these actual *Data* items or *Models* being *Examined*.
Authoritative Data Source	The combination of *Sponsor*, *Examiner*, *Data Source*, and *VV&A Pedigree* which provide one or more *Data* items or *Models* have verified, validated, certified/accredited in accordance with appropriate DoD or Service VV&A procedure. The *Examiner* evaluates that actual *Data* item or *Model* instance provide by the *Data Source* under direction of the *Sponsor* and records these activities in the *VV&A Pedigree.* This definition of *Authoritative Data Source(ADS)* is a compatible extension of the definitions in [2] and [4].

Minimum Requirement:

> All *Models* in CMMS shall be *Registered* by an *Authoritative Data Sources* which have been designated by the appropriate joint and Service specific commands and agencies.

For the military operations *Mission Space*, the actual warfighter (at the appropriate *Level of Warfare*, see Section 2.6) in hostile, live-fire combat operations is the original *Data Source*. However, simulations based on such a *Data Source* are of minimal value in the absence of verification, validation, and eventually accreditation. Doctrine is a disciplined attempt to learn for these warfighter experiences. Just so, the requirement that all *Models* in CMMS be *Registered* by an *Authoritative Data Sources* is an attempt to introduce that same discipline into simulations. CMMS does not conduct VV&A; rather, this Technical Framework provides for the rigorous *Data Source* traceability and configuration management which is required to support subsequent VV&A of CMMS content by competent authority.

There is, however, a pragmatic requirement to balance the natural tension between the delay associated with rigorous VV&A and the immediacy associated with simulation development. This is akin to the maxim that breaking battlefield discipline under duress is a prescription for disaster; but the key to victory is bounded innovation during the battle, under lethal fire. Hence:

Requirement Extension:

> To support concurrent work-in-progress by *Data Sources*, *VV&A Examiners*, and simulation developers, *Models* from *Data Sources* designated by the appropriate joint and Service specific commands and agencies may be *Registered*, *Converted*, and *Integrated* into CMMS in parallel with *Examiner* activities to provide the required VV&A for *Authoritative Data Source* approval.

2.3 Authorized Data Consumers

This section describes the CMMS requirements which control the *Releasability* of any particular *Model* to a specific simulation developer.

Consumer	The combination of a *Person*, *Organization*, and *Role* which constitute the *Receiver* who, because of either mission or subject matter expertise, actually locates, extracts, or evaluates specific CMMS *Data* item or *Model* content, structure, or process for eventual incorporation into a simulation. *Receiver* is defined in Section 2.5 below.
Clearance	The certification that a specific *Consumer* is legally eligible to be entrusted with classified, proprietary, or otherwise sensitive *Data* item or *Model*.
Access	The certification that a specific combination of a *Consumer* with a particular *Clearance* under the authority of an identified *Sponsor* has an appropriate need-to-know for a specific classified, proprietary, or otherwise sensitive *Data* item or *Model*.
Security Pedigree	The comprehensive audit trail which records the specific methods and procedures actually employed by the *Consumer* under authority of the *Sponsor* to ensure that any specific *Data* item or *Model* has been properly protected.
Data Consumer (DC)	The combination of *Sponsor*, *Consumer*, *Clearance*, *Access*, and *Security Pedigree* which requests permission to *Locat*e, *Extract*, or *Evaluate* a specific *Data* item or *Model*. The *Consumer* requests and eventually receives the actual *Data* item or *Model* instance by direction of the *Sponsor* and records these activities in the *Security Pedigree*.
Release Pedigree	The comprehensive audit trail which records the specific methods and procedures actually employed by the *CMMS Manager* under authority of the *Authoritative Data Source (ADS)* to *Release* any specific *Data* item or *Model* to a *Data Consumer*.
Authorized Data Consumer	The combination of *Data Consumer*, *Authoritative Data Source* and *Release Pedigree* certifies the *Release of* one or more *Data* items or *Models* from the *Authoritative Data Source* to the *Data Consumer*. The CMMS Manager records these activities in the *Release Pedigree*. This definition of *Authorized Data Consumer(ADC)* is a compatible extension of the definitions in [2] and [4].

The *Releasability* issue for CMMS is qualitatively and quantitatively more challenging than the security classification and need-to-know requirements usually encountered in simulation development. Following well established systems engineering practice, one of the first steps in simulation development is to decompose the overall simulation problem into a number of well defined simulation components with rigorously defined interfaces. Following the equally well established security classification procedures, the classification level (and corresponding *Releasability*) of specific *Data* items or *Models* are reduced or eliminated precisely because the systems engineering decomposition into components and enforcement of interfaces removes the origin, context, and relationship between *Data* items or *Models* which lead to the original classification levels. However, the CMMS *Conversion* and *Integration* processes are the exact inverse of the usual systems engineering decomposition. CMMS provides the linkage and connection between *Data* items or *Models* which unambiguously identifies the context and relationship and traces the origin which lead directly to higher security classifications and need-to-know restrictions. Therefore CMMS will enforce a strict *Releasability* policy:

<u>Minimum Requirement</u>:

Each *Locator*, *Extractor*, and *Evaluator* shall be an *Authorized Data Consumer*. CMMS shall not *Release* any *Data* item or *Model* to any *Consumer* who is not an *Authorized Data Consumer*.

2.4 Common Syntax and Semantics

A fundamental objective of CMMS is to provide simulation developers with timely and cost-effective access to accurate *Mission* space conceptual *Models* which are created, authenticated, and maintained by others. For example, it is a CMMS objective to enable the direct use of conceptual *Models* of infantry engagements (say developed by WarSim-2000) in the development of close air support conceptual *Models* (say developed by NASM) for eventual use by software developers to implement joint Air-Land Battle simulations (say in JSIMS).

In many cases, the official doctrinally correct language used by the distinct warfare area specialist is a barrier to this direct use and re-use of conceptual *Models* under CMMS. While correct and legitimate within its own domain, the official syntax and semantics in one warfare area often is in direct conflict with the official language in another domain. Even at a basic vocabulary level, there are cases where identical words are used to mean very different things, and there are cases where different words are used to mean the same thing. To make effective use of CMMS, simulation developers require *Mission Space* conceptual *Models* that map domain specific descriptions to a common syntax and semantics.

Moreover, the scope, scale, and diversity of *Models* which CMMS seeks to *Register* includes all Services and every war area operating at each level from national command authorities to individual combatants. This scope, scale, and diversity eliminates manual *Conversion* as a viable option for *Integration*. The specification of *Model* content using common syntax and semantics is essential for any reasonable level of success in automated *Conversion* and *Integration*.

To address these issues, DMSO has formed the Common Syntax and Semantics Technical Working Group (CSS-TWG). The first required step is to define a common language for discussing the needed syntax and semantics:

Vocabulary Atomic concepts which are identifiable and well defined in a standalone fashion, e.g. entities and actions, nouns and verbs. *Vocabulary* is a special case of the more general notion of semantics which seeks to define content and meaning.

Sentence The smallest unit of unambiguous behavior, e.g. subject-verb-object in natural language, domain-mapping-range in mathematics, input-method-output in computer science, entity-action-entity in military missions. *Sentence* is a special case of the more general notion of syntax which seeks to define groups of atomic concepts which produce consistent meaning, see Fig. 2-1.

Context Uniquely identifies the level of abstraction (e.g. level of warfare), internal conditions, or external environment necessary for specific a representation to be selected. *Context* is a special case of the more general notion of meta-data which seeks to define data-about-data.

Relationship The superior/subordinate ordering or ownership/possession specifications necessary to express a particular view of the information as an acyclic graph. *Relationship* is special case of the more general notion of structure which seeks to define connectivity (potentially multi-connected cycles) between syntactic elements.

Canonical Components: The situation where a well-defined set of *Relationships* within a collection of *Sentences* placed in a coherent *Context* yields an atomic concept which can be added to the *Vocabulary*. Air foil, jet engines, control surfaces, and a pilot are all atomic concepts which are identifiable and well-defined in a standalone fashion. Joined together as S*entences* within a C*ontext* and having specified *Relationships*, these become another atomic concept -- manned aircraft -- which can be added to the *Vocabulary*. *Canonical Components* are a special case of the more general notion of normalized, multi-schema representations.

The Technical Framework *Reserve Words* are an example of *Vocabulary*. Examples of *Sentenc*e, *Context*, and *Relationship* will be provided in subsequent sections to define Technical Framework requirements.

Preliminary examinations of representative conceptual *Models* indicate a great diversity in the structural maturity of syntactic form and enforcement of semantic content. The subjective conclusion of this preliminary study is that the utility of a *Model* for CMMS purposes is strongly correlated with structural maturity and enforcement of semantic content, see Figure 2-2:

Internal Knowledge	Subject matter expertise which is available within a particular person or individual. At this level, all structure is implicit (available only within the subject matter expert). Moving from verbal instructions, to memorized procedures, to combat-ready reflex is an *Internal Knowledge* example of increasing semantic content enforcement.
Persistent Natural Language	Written knowledge in the form of *Sentences* and paragraphs. At this level, *Vocabulary* and *Sentence* are made explicit (up the diagram in Fig. 2-2 representing greater structural maturity). Semantic enforcement, for example, moves from *Sentence* fragments on presentation slides, to narrative prose, to structured pseudo-code *Sentences*.
Fully Structured Views	Diagrams which explicitly depict the *Context* and *Relationship* between concepts labeled with *Vocabulary* and described with *Sentences* to specify a particular point-of-view, identifiable perspective, or specific end-purpose. Semantic enforcement increases from IDEF0 activity or process diagrams and IDEF1X entity-relationship diagrams to the unification of entities and actions in RDD-100 behavior diagrams and any number of object-oriented methodologies (Booch, Rumbaugh, Schler-Mellor, ...).
Canonical Representations	The explicit combination of the semantic elements into *Canonical Components* or atomic concepts which are valid specifications for multiple distinct points-of-view, identifiable perspectives, or specific end-purposes.

The relative CMMS *Model* utility/reusability appears to move from lowest in the lower left hand corner to highest in the upper right corner of Figure 2-2. To meet the CMMS *Model* exchange and re-use objectives, specifications and procedures are required to enable CMMS to accept *Models* at various levels of structural maturity and semantic enforcement. And then to *Convert* or migrate those *Models* for I*ntegration* at higher levels of structural maturity and semantic enforcement as shown in Figure 2-3.

Minimum Requirement:

Each *Model* within CMMS shall define its *Vocabulary* in a dictionary.

CMMS shall be capable of *Registering* electronically readable *Models* at each level of structural maturity except Internal Knowledge.

CMMS shall be capable of *Converting Fully Structured Views* and *Canonical Representations* which comply with the appropriate Data Interchange Format (DIF) standards.

Requirement Extensions:

Model developers shall record and *Register Model*-specific *Vocabulary* in an on-line dictionary database system

Model developers shall document and *Register Model*-specific *Sentence*, *Context*, *Relationship*, and *Canonical Component* syntax and semantics.

2.5 Entities, Actions, Tasks, and Interactions (EATI) Representation

There are a number of satisfactory *Fully Structured Views* for describing, displaying, manipulating, or storing a *Model*. The *Model* developer is free to select an appropriate organizing principle and an associated *Fully Structured View* for developing, storing, and displaying the *Model* during *Creation*. However, *Model Conversion* and *Integration* in CMMS is a relatively new concept with little operational experience. To ensure that the *Model* developer provides sufficient information in *Models* to support *Integration*, this section specifies an entities, actions, tasks, and interactions (EATI) representation to define the common syntax and semantics of *Model* content which are independent of an specific *Fully Structured View* approach or CASE tool employed to capture it. The EATI representation is a constraint on the form, content, and usage within all *Fully Structured Views* for the express purpose of supporting *Conversion* and *Integration*.

Entity — A distinguishable person, place, thing, or concept about which information is kept [2]. In particular, *Entity* includes the notions of person, organization, facility, feature, materiel, and plan defined in [5].

State — An *Entity* attribute representing either an internal condition or an external environment.

Event — The location in space and time where a change in S*tate* or condition occurs.

Action — The alteration or transformation by natural force or human agency which produces an *Event*, for example: move, sense, communicate, engage, or replenish.

Role — The function provided by, the part played by, or the character assigned to an *Entity*.

Actor — The *Entity Role*-type which takes, executes, conducts, or controls a particular *Action*.

Supplier — The *Entity Role*-type which sends, constructs, or produces the input of a particular *Action*.

Receiver — The *Entity Role*-type which receives or consumes the output of a particular *Action*.

Direct-Object — The *Entity Role*-type which is generated, transformed, or destroyed a particular *Action*.

Capability	The combination of an Action and a Direct-Object which is recognized as a standard Actor functionality, for example: generate plan, cross river, or shoot missile.
Entrance Criteria	The set of States and the sequence of Events which are necessary and sufficient to initiate, begin, restart, or continue *Action* by an *Actor.*
Exit Criteria	The set of States and the sequence of Events which are necessary and sufficient to terminate, interrupt, end, or conclude *Action* by an *Actor.*
Task	The execution of one or more *Actions* or *Capabilities* by an *Actor.* The Actor initiates execution when specific *Entrance Criteria*. During execution the Action or Capability may receive or consume one or more inputs from *Suppliers*, may produce or deliver to one or more outputs to *Receivers*, and may change one or more Actor *States*. Execution continues until specific *Exit Criteria* are satisfied.
Interaction	The interface which defines the flow of *Events, State*, *Entities,* or *Tasks* between two *Entities or Tasks*.

Some simple *Task* examples are as follows: When attitude reaches 1000 feet (*Entrance Criteria*), aircraft (the *Actor*) reduces climb rate (*Capability*) by 10% per second until level flight is attained (*Exit Criteria*). When directed by JTF Commander (*Event Interaction*), the JFACC (the *Actor*) collects the JTL, the target lists, standard aircraft information, sortie status, and sortie status (inputs via *Interactions* with external *Suppliers*); generates an ATO (*Capability*); and delivers the ATO to all JTF organizations (outputs delivered via *Interactions* with external *Receivers*).

Each EATI component can be recursively decomposed into one more EATI components. For example the Event Interaction above can be decomposed into an *Actor* (the JTF Commander), a Capability (send ATO creation order), and a Receiver (the JFACC) which constitutes a *Task.* Similarly, *Missions* in the EATI representation are *Tasks* and *Interactions* assigned to one or more *Actors* with specific measures of performance. For some Missions, a constituent Task may decomposes that Mission into sub-Missions which are then assigned to other Actors as Tasks with Interactions and performance measures. However, there are restrictions on this recursion of composition and decomposition. For example, an *Entity* can not be assigned as the *Actor* in an *Action* which is based on a *Capability* which the *Entity* lacks.

Minimum Requirement:

CMMS shall not attempt to *Convert* and *Integrate Models* which does not attempt to comply with EATI syntax and semantics.

Each *Model* within CMMS shall provide a dictionary of all *Entities*, *Actions*, *Capabilities*, *Events*, *Tasks*, and *Interactions* employs.

Significant progress has been made in providing common terms for *Entities,* but convergence is not complete. The development of common terms for *Actions* is relatively new. The specific concept of *Task* defined here has not been previously applied to conceptual *Model* specifications of a *Mission*.

Requirement Extension:

CMMS shall be capable of *Integrating* each *Model Created* in compliance with CMMS common syntax and semantics standards described by the EATI template.

The *Model* developer shall employ standard *Entities, Actions, Capabilities, Events, Tasks,* and *Interactions* provided by DMSO where reasonable and practical. The Model developer may deviate from these standard *Entities, Actions, Capabilities, Events, Tasks,* and *Interactions* provided the *Model*-specific deviations are documented in an on-line dictionary system and are mapped to the standard terms.

2.6 Levels of Warfare

Abstraction -- in the form of a hierarchical chain of command and control executing through and across differing organizational levels -- is central to the real structure of and actual behavior in military operations [6]. Combining the levels of warfare and engagement defined in [1, 7], CMMS recognizes the five levels of hierarchical command, control, and execution abstraction in as shown in Figure 2-5 and defined below. Please note: these levels are abstractions introduced by organizations and systems to deal with complexity in the real military operations *Mission Space*. As shown in Figure 2-4, these levels of abstraction are a fact of the *Mission Space* not an artifact of the further abstractions which must be introduced in the actual implementation of simulations.

Strategic Level The level at which national command authorities and combined operational commands determine the security *Objectives*[7] and warfare *Guidance*[7] with associated allocation of *Resources*[7]. With these *Objectives*, *Guidances*, and *Resources*, *Strategic Level* activities establish *Missions*, national and multi-national *Objectives*, sequence initiatives, define limits , and assess risks for the use of military and other instruments of national power. At the *Strategic Level*, the activities lead to the development of global and theater war plans and the provision of military forces and capabilities.

Operational Level The level at which combined operational commands, joint and Service specific task forces plan, conduct, and sustain *Strategic Level Objectives* within geo-spatial theaters of activity. These activities link the *Strategic Level* and the *Tactical Level* by establishing *Objectives*, sequencing *Events*, initiating *Actions*, and applying *Resources* at the appropriate *Operational Level*. *Events* and *Actions* are defined as *Reserve Words* in Section 2.1.2 below.

Tactical Level The level at which joint and Service specific task forces, individual military units, and multi-role platforms plan and execute the ordered arrangement and maneuver of combat elements in space and time relative to own and adversary forces to achieve combat *Objectives.*

Warfighter Level The level at which individual persons, platforms, or combat systems employ sensors, munitions, and communications to remove the combat capability of an adversary

Physical Level The level at which physics, chemistry, biology, or psychology principles are applied to determine material characteristics and performance or to establish required human cognitive and psychological factors.

Level of Warfare One or more of the levels of organizational and engineering abstraction in the command, control, and execution of the real world military operations *Mission Space* defined above.

The first three levels -- *Strategic, Operational, and Tactical* -- are readily recognizable as levels of war defined in [7]. The remaining levels, *Warfighter* and *Physical*, are compatible extensions of the levels in [1] and [7] to individual engagement levels where the details of system operations, engineering characteristics, and human performance are necessary to describe the *Model*.

The *Level of Warfare* defined here provides an example of both *Context* and *Relationship* in the CMMS common syntax and semantics. For a given *Mission*, say a deep penetration air interdiction strike, *Level of Warfare* provides the *Context* for including or excluding details in the *Model* and provides the *Relationships* for defining *Interactions* within that *Model*. In the air strike example, the F-15E aircraft is the same physical aircraft regardless of what *Level of Warfare* the *Model* represents; however, in the real military operations *Mission Space*, the representation presented -- the details which are include or excluded -- to the real warfighter are very different if that warfighter is a general officer in the National Command Authority, a wing operations office in the battle group, a flight leading in the strike package, or the weapon systems officer in the back seat of the aircraft. That is, the real *Mission Space* executes real operations by introducing abstractions which correspond exactly to the *Levels of War* defined above. Hence, explicit identification of these levels is critical *Model* information.

Minimum Requirements:

> Each *Model* within CMMS shall identify the *Levels of Warfare* (and corresponding level of detail) at which that *Model* represents a *Mission*
>
> Each Interaction between two *Tasks* shall specify whether the interface is between components which are peers (at the same *Level of Warfare*) or between superior and subordinate (at distinct *Levels of Warfare*)

The enumerated *Levels of Warfare* provided here are not intended to be exhaustive, but are considered sufficient to span the full range of abstractions which may be required. In particular, some *Models* may refine the *Levels of Warfare* defined here into additional sub-levels of abstraction within the boundaries of one or more of the *Levels of Warfare* defined above.

Requirement Extension:

> Each *Model* which identifies one or more *Levels of Warfare* which are not defined here shall 1.) provide a rigorous definition, and 2.) specify the mapping between the standard levels provided here and the *Model*-specific levels.

3. REFERENCES

1. Under Secretary of Defense for Acquisition and Technology, "Department of Defense Modeling & Simulation Master Plan,", DoD 5000.59-P, October 1995.

2. "DoD Glossary of Modeling and Simulation (M&S) Terms,", DoD 5000.59-M, 29 August 1995.

3. DoD Data Dictionary System, March 1996.

4. DMSO Data Standards Technical Working Group, Authoritative Data Sources Sub-Group minutes, February 1996.

5. GCCS Core Command and Control Data Element Dictionary

6. M.J. Hinich and J.H. Sheehan, "The Necessity of Explicit Levels of Abstraction in M&S Object Models", University of Texas Applied Research Laboratories, 9 March 1996.

7. Universal Joint Task List (UJTL), CJCSM 3500.04, Version 2.1, May 1995.

8. Data Interchange Format, point-of-contact Lt. Col Peter Polk, Defense Modeling and Simulation Office.

9. Modeling and Simulation Resource Repository, point of contact, Gary Misch, Defense Modeling and Simulation Office.

APPENDIX

IDEF1X Entity-Relationship Data Model for the Entities, Actions, Tasks, and Interactions (EATI) Representation.

ATTACHMENT B: CYCLE 1 CMMS DEVELOPMENT PLAN

E-3622U
Development Plan

Functional Area VI - Army-Wide Assessment of MPT Issues

Task Order #09
Development of the Conceptual Model of the Mission Space (CMMS) Prototype Implementation

under the
MANPRINT Support Services (MSS) Contract

Contract Number:
DAAL01-95-C-0115

Date of Preparation: 30 March, 1996

Prepared By:

Dynamics Research Corporation
60 Frontage Road
Andover, MA 01810

Prepared For:

U.S. Army Research Laboratory (ARL)
Aberdeen Proving Ground, MD 21005-5425

Sponsored by:

Defense Modeling and Simulation Office (DMSO)
Alexandria, VA 22311

UNCLASSIFIED

Approvals

Dr. Lawrence H. O'Brien, **Date**
DRC Principal Investigor

Dr. Michael Wagner **Date**
DRC Program Manager

Dr. David Promisel **Date**
ARL COR

Mr. Jack Sheehan **Date**
DMSO CMMS Technical Director

Mark Jefferson, Lt Col, USAF **Date**
DMSO CMMS Program Manager

SECTION 1-INTRODUCTION

1.1 Objective and Document Organization

This document describes the development plan for the Conceptual Model of the Mission Space (CMMS) prototype effort. The plan will be used to guide the development of the CMMS prototype. The CMMS prototype is being developed by Dynamics Research Corporation (DRC) under a task order on the MANPRINT Support Services (MSS) contract.

The development plan will be used by members of the CMMS development team including DRC, its subcontractors, and the Government technical and management personnel. DMSO will distribute an edited version of this specification (without contractor specific financial information) to selected programs and projects, both DMSO managed and externally sponsored, as a template for related activities.

The remainder of the development plan is divided into 6 sections. Section 2 describes the expected target audience for CMMS. Section 3 describes contract deliverables and their expected delivery dates. Section 4 describes the process we will use to accomplish each of the study tasks. Included in this description is a discussion of the software development process and configuration management procedures. Section 5 describes the schedule for completing the study tasks. Section 6 describes the organization structure and resources required to implement the prototype. Section 7 discusses the general management approach that DRC employs on all MANPRINT Support Services (MSS) contracts.

Four technical appendices are provided. Appendix A lists the formats that DRC will use for key deliverables such as the functional and architectural requirements specifications. Appendix B describes the general format used for all MSS reports, including the reports delivered under this task order. Appendix C describes the format for MSS briefings. Appendix D provides a preliminary draft of the questions that will be used during the functional and architectural requirements surveys.

1.2 CMMS Overview

The Defense Modeling and Simulation Office (DMSO) is leading an effort to develop Conceptual Models of the Mission Space (CMMS) as directed in the DoD Modeling & Simulation Master Plan. CMMS is a simulation-independent first abstraction of the real world of activities in a particular mission. The focus of the DMSO CMMS project will be on the military operation mission space. CMMS will provide rigorous specifications of military operations for the simulation developer which are

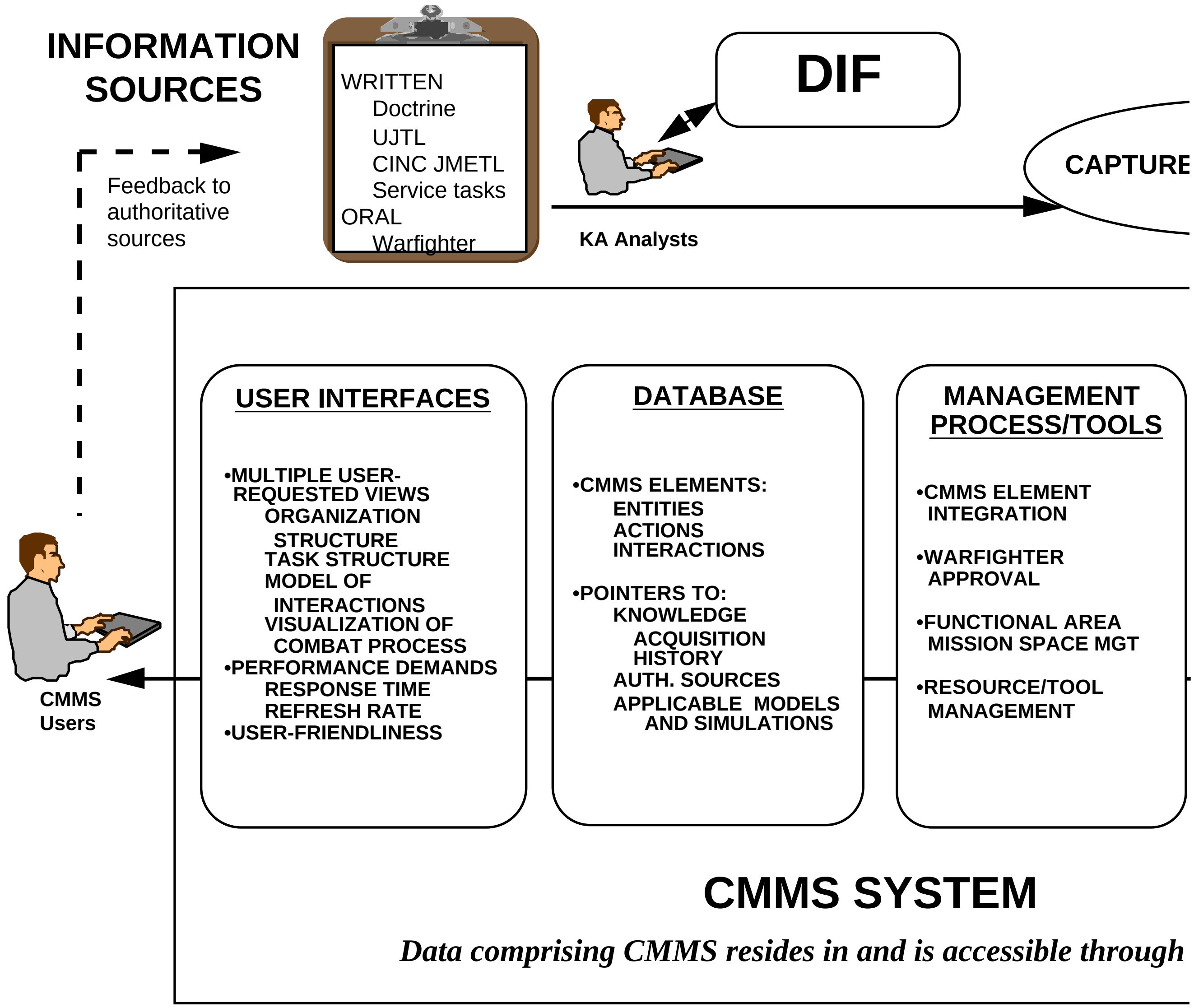
a.) derived from authoritative sources,
b.) described using common syntax and semantics, and
c.) independent of any particular simulation.

CMMS will employ an entities, actions, interactions and transactions template to organize this specification of the military mission space. The source of CMMS is the war fighter; the end-user of CMMS is the simulation developers. CMMS will provide :

1) A hierarchical description of the actions and interactions among the various entities associated with a particular mission area
2) An authoritative first abstraction of the real world
3) A common framework for knowledge acquisition
4) Validated, relevant actions and interactions organized by specific task and entity/ organization
5) A standard format for expression

CMMS seeks to cost-effectively provide simulation-developers (and others) a common understanding of the real world. Figure 1-1 provides an overview of the CMMS system.

CMMS Prime Components



CMMS is a research and development program lead by DMSO. The program had an earlier "Experiments" phase. During the CMMS Experiments phase, three separate experiments were conducted to examine different features of CMMS. Lessons learned from the experiments will be incorporated into the CMMS prototype effort.

Objective Of CMMS Prototype Effort

The objective of the CMMS Prototype effort is to implement a representative, end-to-end thread of the expected CMMS content and a portion of each CMMS-associated process using two iterative development cycles. Defense Modeling and Simulation Office (DMSO) will employ the results of the CMMS Prototype to define the requirements for and the feasibility of a fully operational CMMS system to support major M&S acquisition programs, particularly JSIMS/JWARS, U.S. Army WarSim/FDB, U.S. Navy and Marine Corps knowledge acquisition, Air Force NASM, and ARPA M&S projects such as JWARS-97.

1.3 Applicable Documents

The following documents provided *direct* input data to the development plan:

Document Name	**Revision**	**Date**
Task Description: Development of the Conceptual Model of the Mission Space Prototype Implementation	0	2 April 96
Procedures Manual for MANPRINT Support Services (MSS)	1	12 Feb. 96

Table 1 Related Documentation

SECTION 2-TARGET AUDIENCE

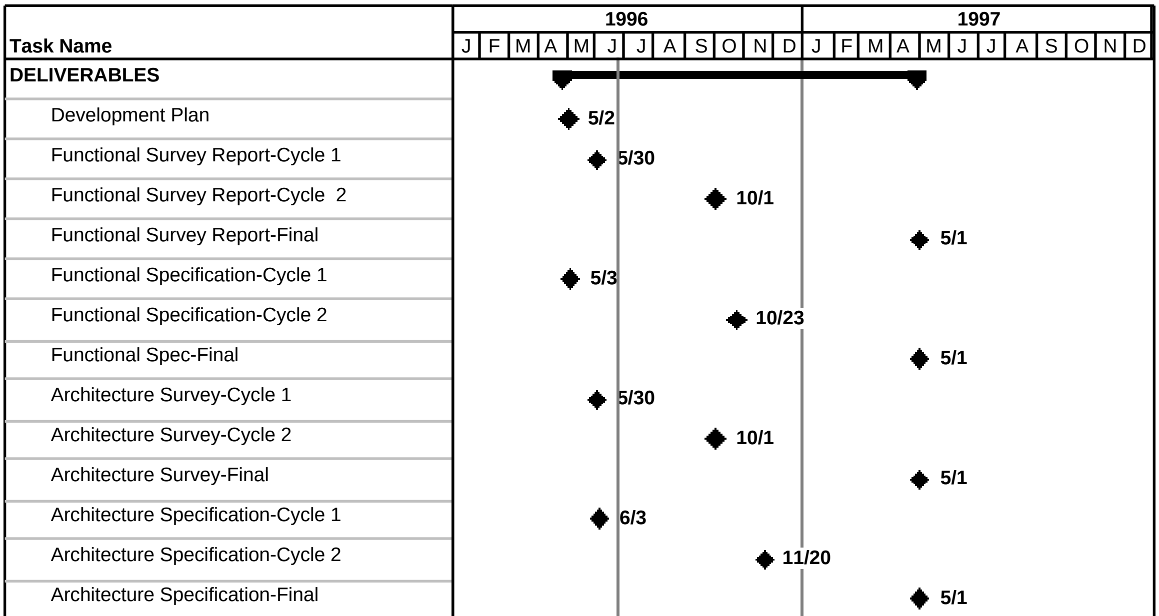
There are three primary types of CMMS end-users

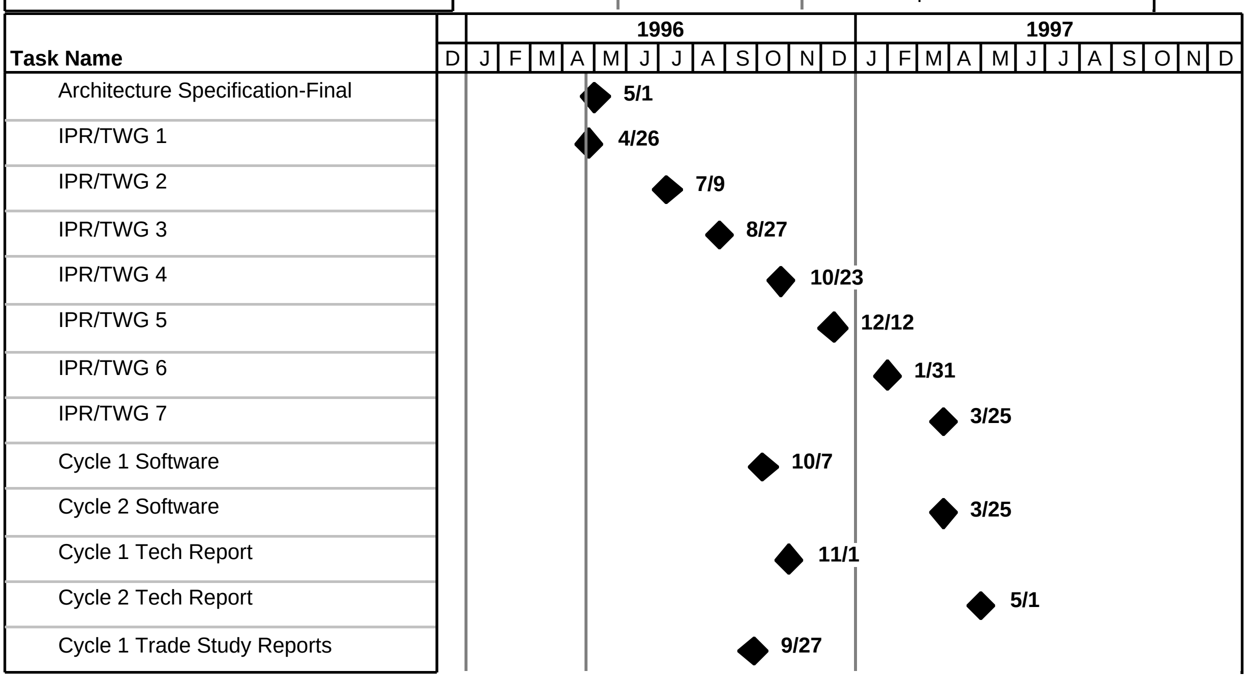
a.) *contributors* who create and register models,
b.) *Data Managers* who integrate, maintain, and release models, and
c.) *simulation developers* who locate and extract models

During the Prototype phase, the contributors will be selected knowledge acquisition specialists from the major M&S programs noted above who will provide previously captured mission space models for registration in the CMMS Prototype. The *Data Manager*s will be the Prototype contractors (DRC) and associated subcontractors (S3I and IMC). The simulation developers will be selected from the major M&S programs to provide Prototype requirements and to evaluate the Prototype functionality.

SECTION 3-DELIVERABLES

The task order deliverables and their associated delivery dates are displayed below.





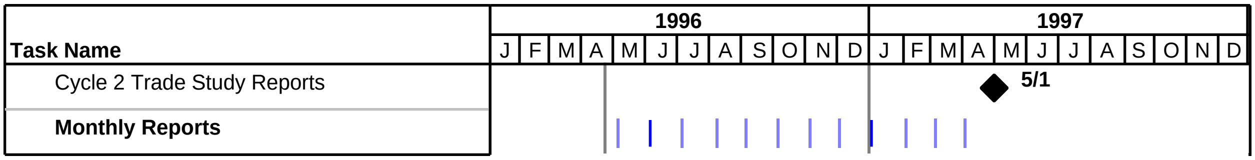


Table 3-2 lists the magnetic storage medium that will be used for each deliverable. One printed copy and one magnetic storage media copy will be provided for each deliverable. Formats for each deliverable are described in Section 4.

MAGNETIC MEDIUM	**DELIVERABLE**
Development Plan	Word For Windows 6
Function Survey Reports	Word For Windows 6
Functional Specifications	Word For Windows 6
Architecture Survey Reports	Word For Windows 6
Architecture Survey Specifications	Word For Windows 6
IPR/TWGs	PowerPoint 4
Trade Study Reports	Word For Windows 6
Software -Users Manual	Word For Windows 6
Software-Code	(Magnetic Storage Only)
Technical Reports	Word For Windows 6
Monthly Reports	Word For Windows 6

SECTION 4-DEVELOPMENT APPROACH

4.1 Assumptions, Dependencies, and Restrictions

1) CMMS Technical Framework. To successfully develop a functional requirements specification for the CMMS prototype that is consistent with the CMMS Technical Framework, DRC must be provided with a copy of the CMMS Technical Framework at least 20 days before the functional specification will be delivered.

2) Access To Simulation Developers. During the requirements surveys and the prototype demonstrations, DRC will need access to simulation developers from key Government programs. DRC will need approximately 6-14 hours during both the surveys and the demonstrations.

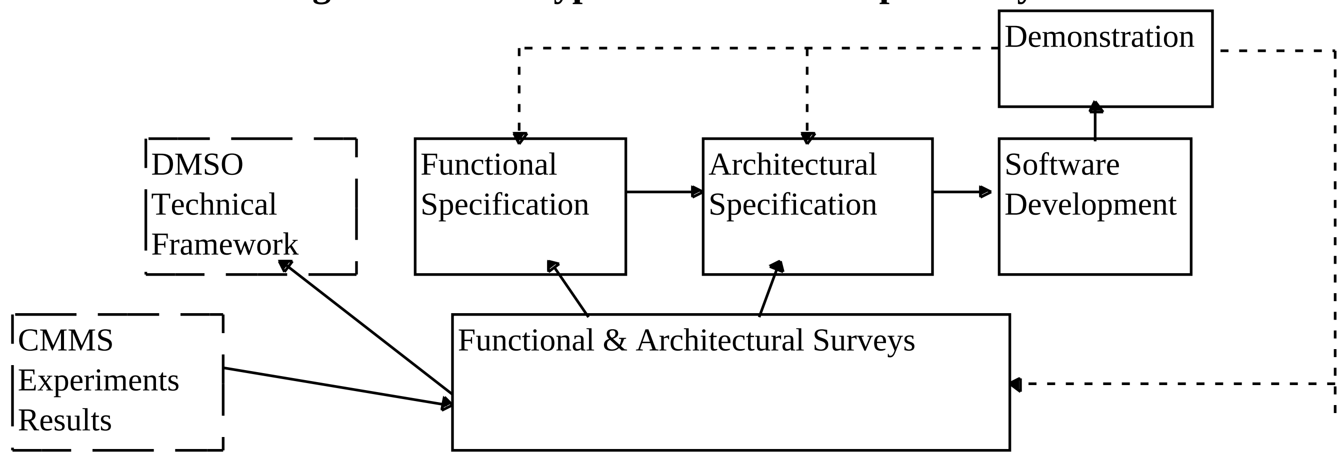
3) Inputs from MSSR. It is assumed that the MSSR will provide the mechanisms for users to access CMMS. Based on this, it is further assumed that the MSSR will provide a number of services associated with data access such as user registration. The specific services which are expected to be obtained from the MSSR will be identified in the CMMS functional specification. Timely delivery of these services will be need to demonstrate a full range of CMMS capabilities.

4) Externally-Derived Semantics and Syntax. If DRC is expected to incorporate an "externally" derived semantics and syntax structure (i.e., a structure developed outside of DRC's task order) into the CMMS common data base, DRC must receive that data structure no later than 30 days after the start of the task order.

4.2 Development Cycles

The CMMS prototype will be developed in two six-month cycles which will be called Cycle 1 and Cycle 2. At the beginning of each cycle, simulation developers will be surveyed to identify potential functional and architectural requirements (see Figure 4-1). Functional requirements will be documented in a functional specification delivered 30 days after the start of each cycle. Results of the requirements surveys will be documented in reports delivered 60 days after the start of each cycle. Once the functional requirements specification for a cycle is approved, the architecture specification for that cycle will be developed and delivered 30 days later. Software (code plus user's manual) will be delivered 6 months after the beginning of each cycle. The software will then be demonstrated to users. Feedback from the demonstration will be used to update the functional and architecture requirements specifications.

Figure 4-1 Prototype Software Development Cycle



4.3 Work Breakdown Structure

Table 4-1 provides lists the WBS for the project. The high-level tasks in the WBS are derived from the task order SOW tasks.

Table 4-1 WBS

WBS #	WBS NAME
1.0	Develop Development Plan
2.0	Develop Functional Requirements Specifications
2.1	Cycle 1-Functional Requirements Specifications
2.1.1	Cycle 1- Develop Functional Requirements Specifications
2.1.2	Cycle 1- Conduct Functional Requirements Survey
2.2	Cycle 2- Functional Requirements Specifications
2.2.1	Cycle 2- Develop Functional Requirements Specifications
2.2.2	Cycle 2- Conduct Functional Requirements Survey
3.0	Develop Architectural Requirements Specifications
3.1	Cycle 1-Architectural Requirements Specifications
3.1.1	Cycle 1- Develop Architectural Requirements Specifications
3.1.2	Cycle 1- Conduct Architectural Requirements Survey
3.2	Cycle 2-Architectural Requirements Specifications
3.2.1	Cycle 2- Develop Architectural Requirements Specifications
3.2.2	Cycle 2- Conduct Architectural Requirements Survey
4.0	Produce Technical Reports
4.1	Cycle 1 Technical Report

4.2	Cycle 2 Technical Report
5.0	Collect/Identify Knowledge
5.1	Cycle 1 Knowledge Identification
5.2	Cycle 2 Knowledge Identification
6.0	Conduct Trade Studies
6.1	Cycle 1 Trade Studies
6.2	Cycle 2 Trade Studies
7.0	Develop Prototypes
7.1	Cycle 1 Prototype Development
7.1.1	Develop Common Data Base Structure
7.1.2	System Integration & Testing
7.1.3	CSCI A
7.1.4	CSCI B
7.1.5	CSCI C
7.1.x	CSCI x (as required)
7.2	Cycle 2 Prototype Development
7.2.1	Develop Common Data Base Structure
7.2.2	System Integration & Testing
7.2.3	CSCI A
7.2.4	CSCI B
7.2.5	CSCI C
7.2.x	CSCI x (as required)
8.0	Demonstrate Prototype
8.1	Cycle 1 Prototype Demonstration
8.2	Cycle 2 Prototype Demonstration
9.0	Management
9.1	Support IPRs/TWGs
9.2	Provide Monthly Reports
9.3	Provide Technical Management
9.4	Purchase Materials
9.5	Travel

4.4 Approach to Individual Tasks

This section describes the process that will be used to accomplish each task described in the SOW.

4.4.1 Task 1: Produce Development Plan

This document describes the development plan.

4.4.2 Task 2: Develop Functional Specifications

4.4.2.1 Develop Functional Requirements Specifications

Appendix A-1 provides a proposed outline of the contents of the functional requirements specification. The initial functional requirements specification due at the beginning of Cycle 1 will be derived from: (1) CMMS Technical Framework, (2) CMMS Prototype SOW, and (3) results from the user functional requirements survey.

The second functional requirements specification delivered at the beginning of Cycle 2 will be derived by updating the initial specification to reflect results of the Cycle 1 demonstration and Cycle 2 requirements survey.

In both versions of the functional requirements specification, requirements will be annotated to identify which requirements will be implemented in Cycles 1 and 2, and which will be implemented in the operational CMMS system.

4.4.2.2 Conduct Functional Requirements Survey

A small select group of simulation developers will be interviewed at the beginning of each development cycle to obtain inputs on CMMS functional and architectural requirements. A draft of the questions to be used in the Cycle 1 interviews is listed in Appendix D. The primary objective of the Cycle 1 surveys is to engage the major M&S development programs in the definition CMMS. As a minimum, selected knowledge acquisition contributors and simulation developers from the following programs will be visited during the Cycle 1 interviews:

JSIMS-Orlando, FL
JWARS--Orlando, FL
NASM--Bedford, MA
WARSIM--Orlando, FL
Navy Simulation Programs--Norfolk, VA.

The Cycle 2 survey will employ a mail-survey approach to survey a wider range of simulation developers. The Cycle 2 survey has the additional objective of extracting functional requirements and system architecture from existing “best of breed” knowledge acquisition experts and simulation developers who may not be associated with the above programs. These programs will be identified within 60 days before the start of Cycle 2.

4.4.3 Task 3: Develop Architecture Specifications

4.4.3.1 Develop Architecture Specifications

Appendix A-2 provides a proposed outline of the contents of architectural requirements specification. The initial architectural requirements specification due at the beginning of Cycle 1 will be derived from: (1) CMMS technical framework, (2) Cycle 1 functional requirements specification, and (3) results from the Cycle 1 architectural requirements survey.

The second specification delivered at the beginning of Cycle 2 will be derived by updating the initial specification to reflect results of the Cycle 1 demonstration and Cycle 2 architectural requirements survey.

In both versions of the architectural requirements specification, requirements will be annotated to identify which requirements will be updated in Cycles 1 and 2 and which will be implemented in the operational CMMS system

4.4.3.2 Conduct Architecture Requirements Survey

The same group of simulation developers who will be interviewed or surveyed for the requirements survey will also be contacted to obtain input on both their current architecture for storing warfighter knowledge and their preferences for a CMMS architecture. A draft of the questions to be used in the Cycle 1 survey is listed in Appendix D.

4.4.4 Task 4: Prepare Technical Reports

A brief technical report (20-30 pages) will be developed to document lessons learned from each software development cycle. These reports will follow the general MSS report format which is listed in Appendix B.

As part of this task, DRC will coordinate development of the CMMS prototype with development of the High Level Architecture (HLA) for the Joint Mission Space Model (JMSM), a component of the Joint Simulation System (JSIMS).

4.4.5 Task 5: Identify/Collect Existing Knowledge

The goal of the CMMS prototype is to demonstrate the full range of CMMS functionality using a “thread” or slice of knowledge that is representative of the CMMS content. During this task, the thread will be identified and the knowledge needed to support the thread will be collected from existing simulation programs. In developing the thread, the goal will be to maximize use of existing knowledge. No additional knowledge acquisition will be undertaken without the prior approval of the Government program manager.

DRC's current plan is to use a Joint Interdiction operation as the top of the thread. This operation will be broken down into a series of lower level actions including Army, Air Force, and Navy Air Interdiction missions as well as land interdiction, SOF interdiction, and sea interdiction missions. These lower level missions should provide the hooks needed to link with existing simulations such as JSIMS, NASM, and WARSIM.

4.4.6 Task 6: Conduct Trade Studies

Trade studies will be conducted on an as required basis to evaluate functional or architectural alternatives. Each trade study will be documented in brief (2-3 page) report describing the following:
-Statement of Problem or Issue
-Description of Alternatives
-Criteria for Evaluation
-Criterion Scores For Each Alternative
-Recommendation

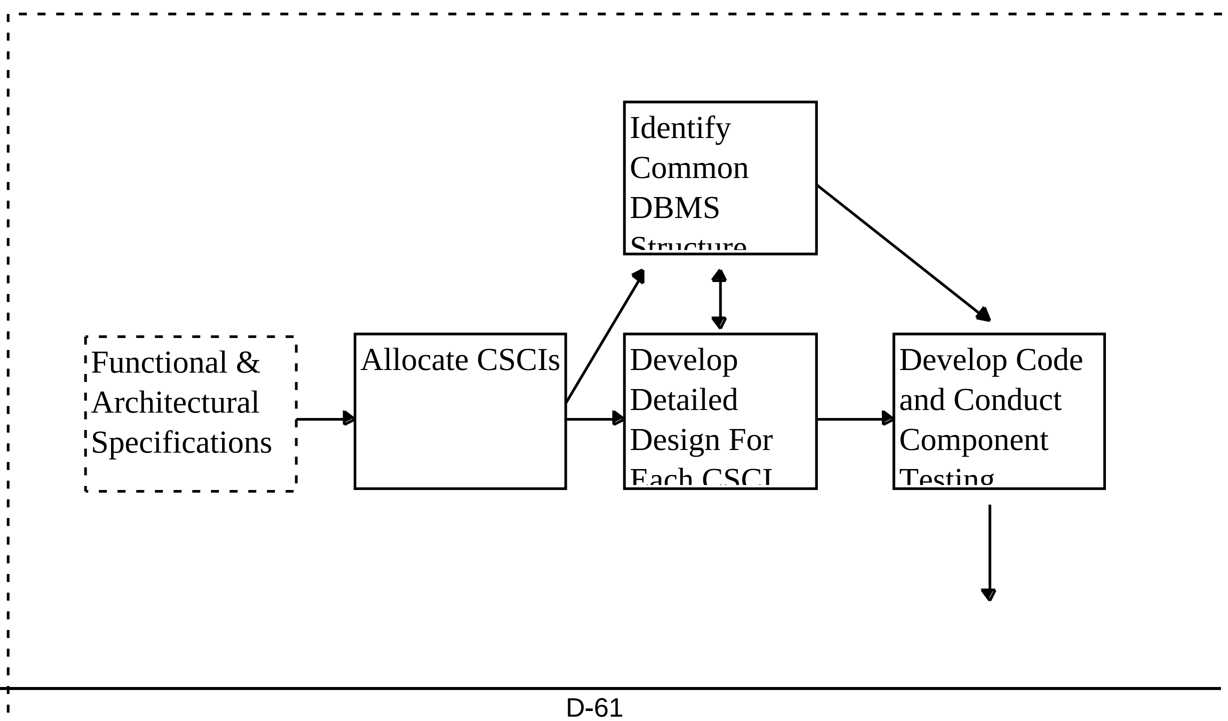
Trade study reports will follow the general format used by all MSS reports. This format is described in Appendix B.

4.4.7 Task 7: Develop Prototype

4.4.7.1 Software Development Process

Because this is a prototype effort, DRC will employ a simplified version of its software procedures. This major steps in this process are depicted in Figure 4-2.

Figure 4-2 Software Development Process



Inputs from Requirements Specifications. During the development of the architecture specification, we will identify : (1) the CMMS Computer Software Configuration Items (CSCIs) and their major relationships, (2) target (or run-time) hardware and software components, and (3) the software development environment (hardware and software) which will be used during prototype development.

Allocation Of CSCIs. Once the architectural specification has been approved, individual CSCIs will be allocated to members of the two software development teams (DRC and S3I). The assigned team will then have responsibility for detailed design and component testing and debugging of the CSCI, fixing bugs related to the CSCI identified during system test and integration, and for writing the associated section of the User Manual.

Identification of Common DBMS. CMMS will have two major types of software items--those that convert simulation program data into a common data base and those that search for and extract data from this data base. In either case, the common data base plays a central role in the development of the CMMS prototype. Development of individual software items cannot proceed far until the common data base structure had been identified.

Immediately after the functional requirements specification has been approved, a select team will attempt to identify the major data elements to be included in the common data base. Inputs for this identification process will be: (1) the requirements interview results, (2) externally developed CMMS semantics and syntax, and (3) CMMS experiment data bases and concepts. Initially, the team will identify major data elements (i.e., table names) and their relationships. Then the team will identify individual data elements (data fields) and their attributes (e.g., data types and sizes) within each table. To facilitate the rapid development of a data structure, a relational data base framework will be used to describe the data base structure. A COTS package, ERWIN, will be used to document the data structure in IDEF1X format. Within this relational framework, common symantical terms will be stored in “library” files. A data element will then be associated with the items in these library files to describe the element’s place in the CMMS semantic framework. For example, a task may be linked to elements in both the UJTL and common verb libraries. Elements of the common syntactical structure will be represented via data base table and field elements and relationships.

Once a common data base structure has been identified, the data base will be implemented using an COTS DBMS. In addition, library files containing the common semantic elements will be populated.

To successfully meet the development schedule, the goal will be to complete development of the common data base within 30 days of approval of the architecture specification.

Once the common data base has been baselined, it will be placed under configuration control and no changes to it may be made without the approval of the DRC principal investigator.

Develop Detailed Design. Concurrently with the development of the common data base structure, a detailed design will be developed for each CSCI. The detailed design will describe: (1) CSUs or lower level components and their key attributes and methods, (2) data flow including inputs and outputs to the CSCI and flow among CSUs, (3) execution control flow, and, where applicable, (4) user computer interface design. The latter will include static prototypes of individual screens.

Coding of individual CSCIs will not begin until the detailed design has been approved by the DRC principal investigator. To obtain this approval for complex CSCIs, a “design walkthrough” may be required.

Code and Component Testing. CSCIs will be developed by the S3I and DRC software teams at their respective sites. Code must be developed in accordance with the general coding standards identified in the architecture specification. These standards will describe general procedures for structuring, building, and documenting code. For the most part, it is expected that CSCIs will be independent applications that work off a common data base. Thus, integration and configuration management issues should be minimized. A COTS tool will be used to provide configuration management capabilities for all software development. DRC will use this software to maintain up-to-date versions of each CSCI on a server that will be dedicated to CMMS. S3I will provided with access to this server to download/upload software items. Only the lead for each CSCI may upload a modification of a software item. Once development is initiated, each lead will be expected to perform an upload each week. All members of the team may download the latest version of each CSCI.

Each CSCI team will be responsible for component testing of that CSCI. The initial focus of that testing will be on functional requirements allocated to the CSCI in the architecture specification and design features described in the detailed design description. Once the demonstration script has been developed (see below) component testing may be initiated. Component testing will focus on direct testing of the CSCI items included in the script.

Initial CSCI coding and component testing of all CSCIs must be completed at least two weeks prior to completion of the software development cycle to allow sufficient time for system integration, implementation, and testing.

System Integration, Implementation, and Test. DRC will perform all builds involving integration of more than one CSCI. Two months after the architecture specification is approved, a demonstration script will be developed. The script will describe the functional features and data elements to be demonstrated at the end of the software cycle. The focus of system testing will be on a "walkthrough" of this script to ensure that all required elements can be successfully demonstrated. The functional features to be demonstrated will be obtained from the requirements specifications. The data elements to be demonstrated will be obtained from the "thread" identification task. As part of the walkthrough process, selected CSCIs will be applied and the date base needed for the demonstration will be populated.

4.4.8 Task 8: Demonstrate Prototype

After software for each cycle has been delivered, it will be demonstrated to the Government CMMS staff as part of an IPR. It will then be demonstrated to end users in conjunction with a TWG. The software will then be demonstrated at approximately 5 key simulation developer sites. The Cycle 1 demonstration will be conducted in conjunction with the Cycle 2 requirements surveys. The focus of these surveys will be on obtaining feedback on the functions represented in the Cycle 1 software.

4.4.9 Task 9: Management

Section 7 describes general management procedures applicable to all MSS task orders and deliverables. The remainder of this subsection describes management-related processes related to unique CMMS management subtasks.

4.4.9.1 Support IPRs/TWGs

DRC will provide support to a maximum of seven IPRs and seven TWGs. It is assumed that the IPR will be held on the day preceding each TWG. Briefing slides for the TWG will be previewed at the IPR. Slides will be developed in accordance with the format described in Appendix C. The slides will address overall project schedule and progress as well as issues requiring attention or action by TWG members. Software demonstrations will be provided at those TWGs immediately following delivery of software for each development cycle.

It is assumed that the IPRs will be informal in nature and will focus on technical issues, progress, problems, and actions requiring resolution or attention by DMSO. Separate briefing slides will not be developed for the IPRs.

4.4.9.2 Prepare Monthly Reports

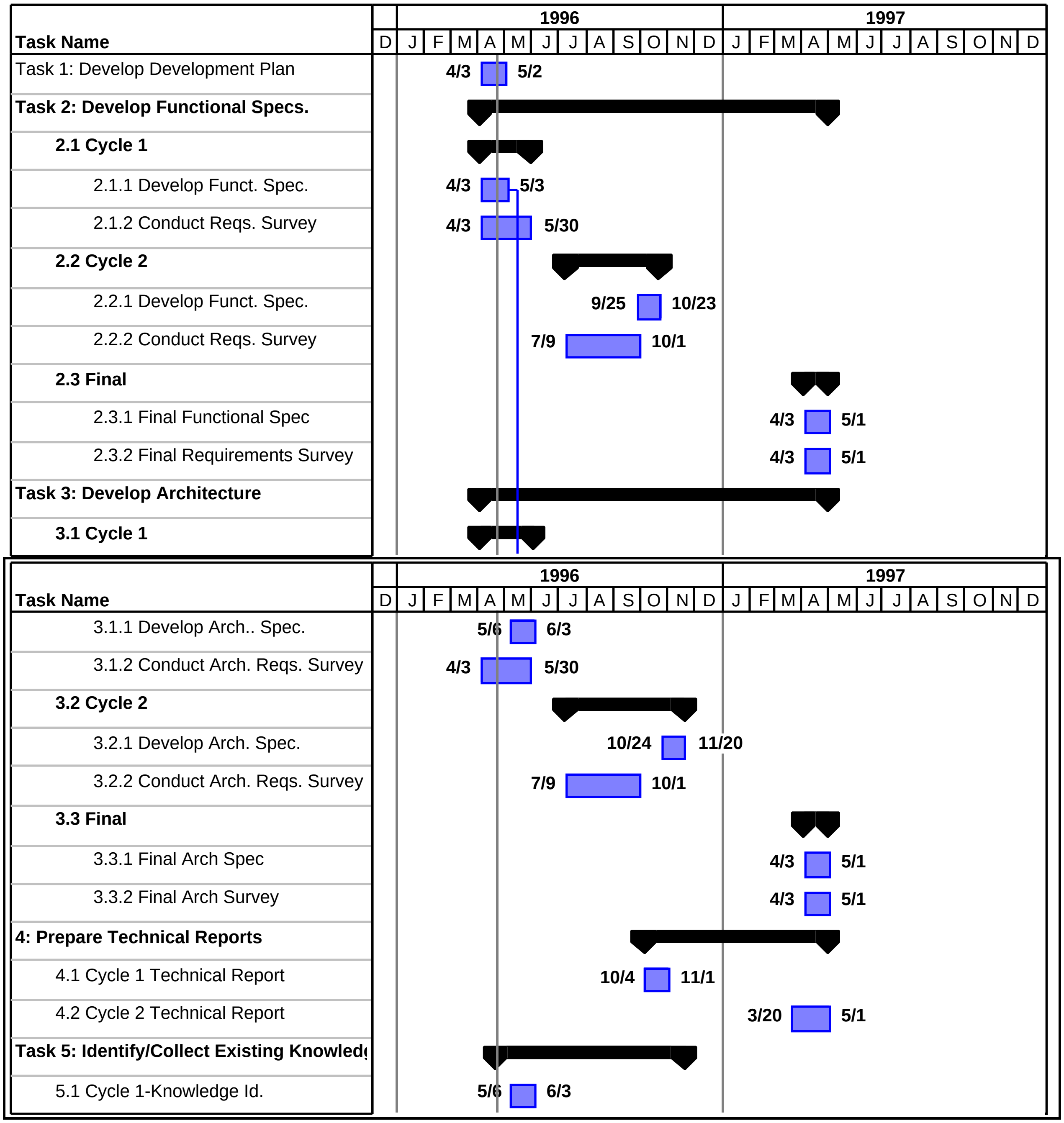
The procedure and format for delivering the monthly report will be the same as that used for all MSS task orders. DRC will deliver the monthly reports to ARL 10 days after the start of each DRC fiscal period. This format is described in Section 7.3.

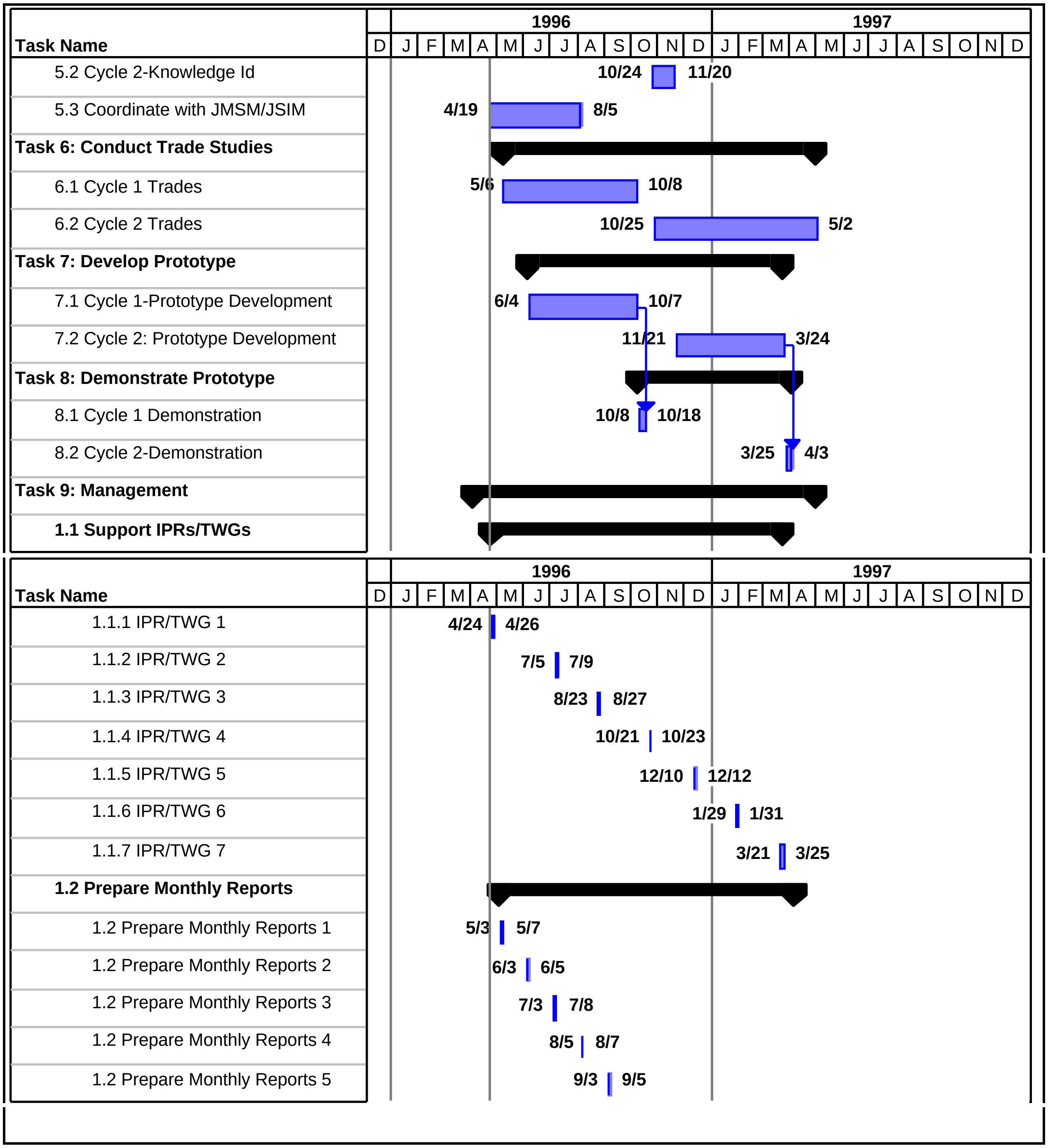
4.4.9.4 Technical Management

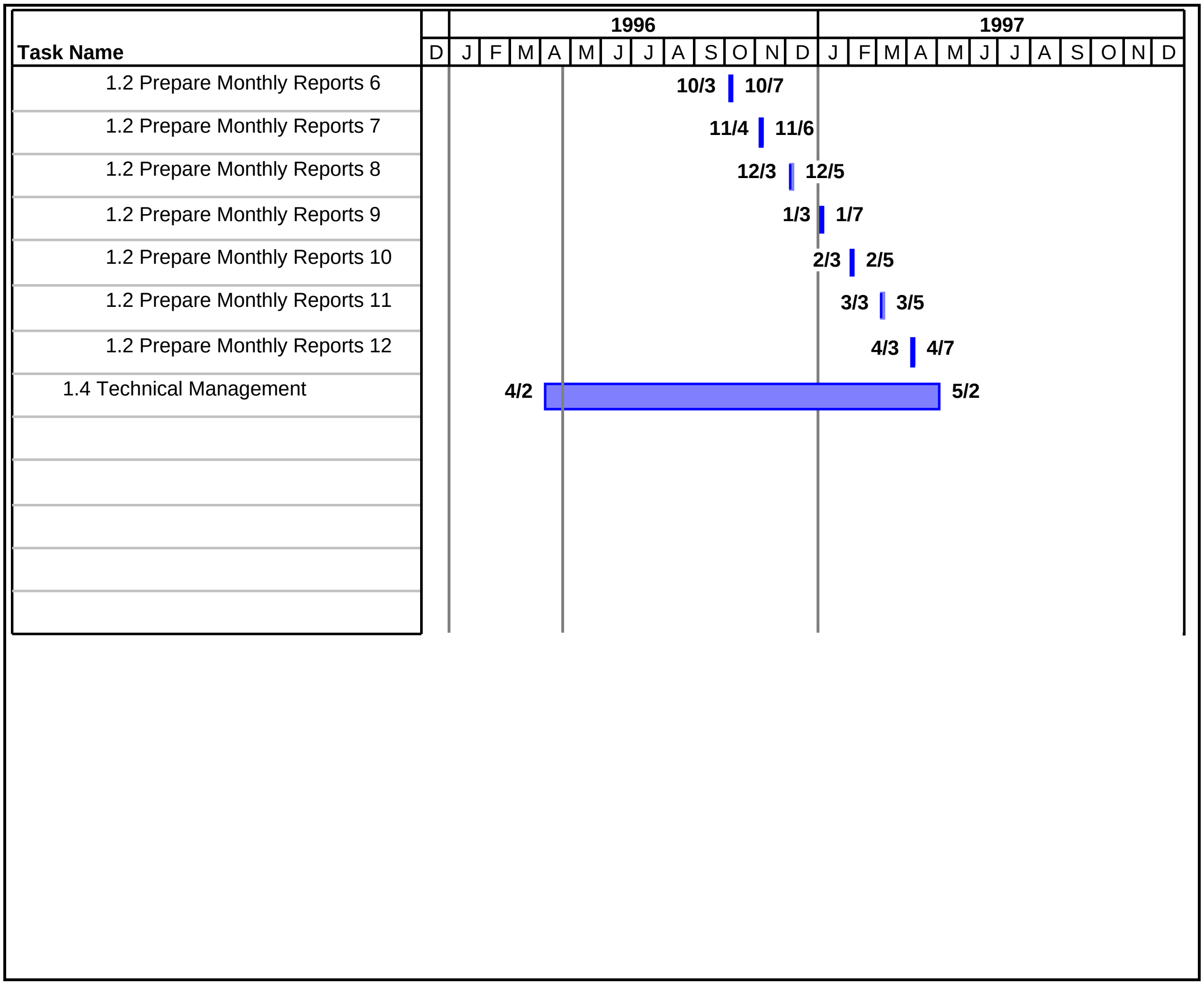
SECTION 7.1 DESCRIBES THE TECHNICAL MANAGEMENT APPROACH USED ON ALL MSS TASK ORDERS. SECTION 6 DESCRIBES ORGANIZATIONAL RESPONSIBILITIES FOR THE CMMS PROTOTYPE EFFORT.

SECTION 5-SCHEDULE

Figure 5-1 lists the schedule for the study tasks.




Task Name
1996
1997
D J F M A M J J A S O N D J F M A M J J A S O N D
5.2 Cycle 2-Knowledge Id
10/24
11/20
5.3 Coordinate with JMSM/JSIM
4/19
8/5
Task 6: Conduct Trade Studies
6.1 Cycle 1 Trades
5/6
10/8
6.2 Cycle 2 Trades
10/25
5/2
Task 7: Develop Prototype
7.1 Cycle 1-Prototype Development
6/4
10/7
7.2 Cycle 2: Prototype Development
11/21
3/24
Task 8: Demonstrate Prototype
8.1 Cycle 1 Demonstration
10/8
10/18
8.2 Cycle 2-Demonstration
3/25
4/3
Task 9: Management
1.1 Support IPRs/TWGs
Task Name
1996
1997
D J F M A M J J A S O N D J F M A M J J A S O N D
1.1.1 IPR/TWG 1
4/24
4/26
1.1.2 IPR/TWG 2
7/5
7/9
1.1.3 IPR/TWG 3
8/23
8/27
1.1.4 IPR/TWG 4
10/21
10/23
1.1.5 IPR/TWG 5
12/10
12/12
1.1.6 IPR/TWG 6
1/29
1/31
1.1.7 IPR/TWG 7
3/21
3/25
1.2 Prepare Monthly Reports
1.2 Prepare Monthly Reports 1
5/3
5/7
1.2 Prepare Monthly Reports 2
6/3
6/5
1.2 Prepare Monthly Reports 3
7/3
7/8
1.2 Prepare Monthly Reports 4
8/5
8/7
1.2 Prepare Monthly Reports 5
9/3
9/5


Task Name
1996
1997
D J F M A M J J A S O N D J F M A M J J A S O N D
1.2 Prepare Monthly Reports 6
10/3
10/7
1.2 Prepare Monthly Reports 7
11/4
11/6
1.2 Prepare Monthly Reports 8
12/3
12/5
1.2 Prepare Monthly Reports 9
1/3
1/7
1.2 Prepare Monthly Reports 10
2/3
2/5
1.2 Prepare Monthly Reports 11
3/3
3/5
1.2 Prepare Monthly Reports 12
4/3
4/7
1.4 Technical Management
4/2
5/2

SECTION 6 ORGANIZATIONAL STRUCTURE AND RESOURCES

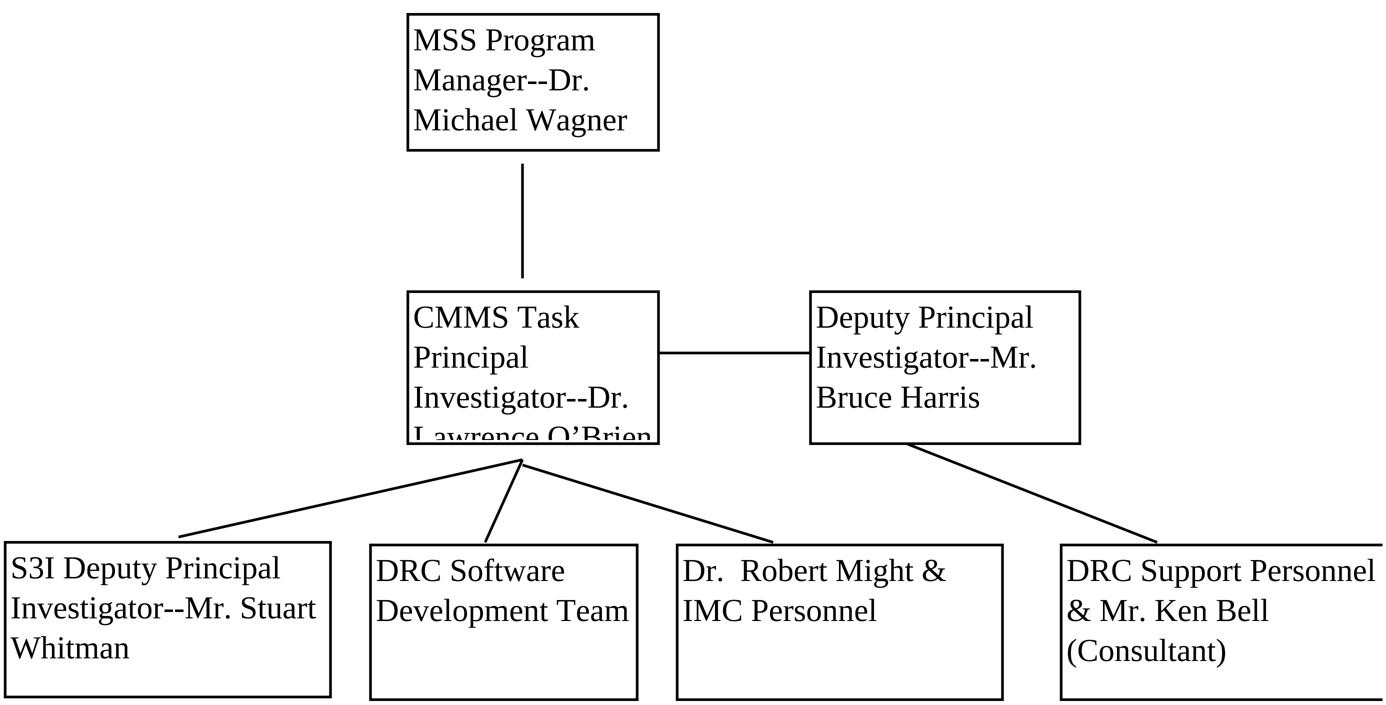
DRC will be the prime contract for the CMMS prototype development effort. Subcontractors on the effort will be S3I and IMC.

6.1 Staffing Plan and Manpower Loading

[Contractor specific financial information, contact Dr. Larry O'Brien at DRC for further information.]

6.2 Organizational Structure

Figure 6-1 provides an overview of the project organizational structure. Dr. Michael Wagner is the Program Manager for the MSS task order contract. Dr. Lawrence O'Brien will be the Principal Investigator for the CMMS prototype effort and, as such, will have primary responsibility for both technical and staffing decisions. Dr. O'Brien will be assisted by Mr. Bruce Harris from DRC, Stuart Whitman from S3I, and Dr. Bob Might from IMC. Mr. Harris will assist Dr. O'Brien on management tasks such preparation of deliverables (e.g., IPRs/TWGs, monthly reports) and will assist in the requirements survey. He will also have primary responsibility for identifying and collecting data on the warfighting thread to be demonstrated in the prototype. Mr. Ken Bell, a consultant, will coordinate with JSIMS ongoing knowledge capturing efforts in Orlando. Mr. Whitman will direct S3I efforts on the prototype while Dr. Might will direct IMC's efforts.



S3I Software Development Team

Two software engineering teams, one at DRC and one at S3I, will develop the CMMS prototype software. Clerical and support personnel from all three companies will provide support for the development of project reports and briefings.

6.3 Materials

Hardware and software will probably need to be purchased to support unique CMMS development requirements. The specific hardware and software to be used in the development suite will be identified in the architecture specification. At that time, more specific requirements for material purchases will be identified.

6.4 Travel

Table 6-3 lists the trips required to support the CMMS prototype effort.

OBJECTIVE	**# Trips**	**COMPANY**	**SOURCE**	**DESTINATION**	**# OF PERS.**	**# OF DAYS**
IPR/TWG	7	DRC	Boston	Washington, DC	2	2
Reqs. Survey-WARSIM/JSIMS/JWARS	3	DRC	Boston	Orlando, FL	3	5
Reqs. Survey-WARSIM/JSIMS/JWARS	3	IMC	Wash. DC	Orlando, FL	2	5
Reqs. Survey-WARSIM/JSIMS/JWARS	3	S3I	Wash. DC	Orlando, FL	2	5
Reqs. Survey--Norfolk	2	DRC	Boston	Norfolk, VA	2	3
Reqs. Survey--Norfolk	2	S3I	Wash. DC	Norfolk, VA	2	3
Reqs. Survey--Norfolk	2	IMC	Wash. DC	Norfolk, VA	2	3
Reqs. Survey--NASM	2	IMC	Wash. DC	Boston, MA	2	3
Reqs. Survey--NASM	2	S3I	Wash. DC	Boston, MA	2	3
Management Coord.	3	DRC	Boston	Wash. DC	2	1
Management Coord.	2	S3I	Wash. DC	Boston, MA	1	1
Management. Coord	4	DRC	Boston	Orlando, FL	1	2
Data Collection[9]	4	DRC	Boston	Ft. Leavenworth, KA	1	3

[9] This Fort Leavenworth site is used a place-holder for additional data collection sites because of its central location.

SECTION 7 GENERAL MANAGEMENT ISSUES

7.1 TECHNICAL AND ADMINISTRATIVE MANAGEMENT

[Contractor specific management information, contact Dr. Larry O'Brien at DRC for further information.]

APPENDIX D: DRAFT FUNCTIONAL AND ARCHITECTURAL REQUIREMENTS SURVEY QUESTIONS

Part I-Background Information

Part IA-Program Information

Program Name: ________________________ **Proponent**: ______________________
Development Agency: ___________________

SIMULATION OBJECTIVES (TYPE OF OBJECTS/ENTITIES SIMULATED):

TARGET AUDIENCE:

Milestone Dates:

ORD ________________
INITIATION OF KNOWLEDGE CAPTURING: ________________
COMPLETION OF KNOWLEDGE CAPTURING: ________________
DEVELOPMENT CONTRACT AWARD: ________________
IOC_____________
FOC_____________

Part IB-Respondent Data

NAME	*PROGRAM ROLE*	*ORGANIZATION*	*PHONE*	*E-MAIL*

Part II-Functional Requirements Survey

Part IIA-Knowledge Capturing Tools

1) What tools (e.g., CASE tools, DBMSs) do you use (or plan to use) to capture warfighting knowledge ?

2) Are you happy with the tool(s) ? ____ YES _____ NO _______________________________

3) What are the tool's best features ?

4) What are the tool's worst features ?

5) What format does the tool use to store data ?

6) Does the tool have capabilities to export to data to other formats ? If so, what are those formats ?

7) How do you distribute the knowledge captured in the tool for review or validation ?

8) Does the tool have a run-time viewer ? If so, do you use it ? Do you have any information on it ?

9) How many people will use the tool(s) to:
_____(a) capture knowledge
_____(b) review/validate knowledge

10) How frequently is the information in the tool:

_____(a) updated,
_____(b) distributed for review ?

10) Do you captured classified or proprietary data ? If so, what tools/techniques do you use to deal with classified data ?

11) What tools and techniques do you use to handle changes in the data sources associated with the knowledge ?

12) Do you have a delimited set of semantic terms (e.g. a taxonomy or classification scheme for organization types or tasks) ? If so, could we obtain documentation on them ?

13) Does your tool require knowledge be described with a specified syntactical structure ? If so, could we obtain documentation on the structure ?

14) Does your knowledge base have an object or data model ? If so, could we obtain documentation on it ?

Part IIB-Knowledge Content

Assume you were building a new simulation and you had $1000 to spend to obtain 'real world" knowledge on the warfighting slice you had to simulate. Show how you would spend the $1000 by dividing it among the different types of knowledge listed below. You must spend the entire $1000. You can distribute the money among the choices in any manner you wish--you do not have "buy" some of everything ? For any knowledge type where you would spend money also indicate if your are currently capturing knowledge on that type of knowledge in your simulation program.

$$$ In SIM (Y/N)

_____ ______ List of organization types (example organization type: Armor Battalion)
_____ ______ Breakout or decomposition of organization types into subordinate organization types
_____ ______ List of tasks performed by organization type
_____ ______ List of operations or missions performed by organization type
_____ ______ List of tasks associated with each operation
_____ ______ Breakout or decomposition of tasks into lower level tasks
_____ ______ Information flow (inputs/outputs) between tasks
_____ ______ Sequence of performing tasks during an operation or mission
_____ ______ Sequence of performing subtasks within a "parent" task
_____ ______ Task initiation, termination, and interrupt cues
_____ ______ Assignment of tasks to specific organization types
_____ ______ Assignment of tasks to individual personnel positions
_____ ______ Doctrinal references providing more detailed description of each operation
_____ ______ Doctrinal references providing more detailed description of each task
_____ ______ Assignment of higher-level tasks to UJTL
_____ ______ Communication links (i.e., communication channels and systems) between organization types
_____ ______ Communication or message *protocols* used by an organization type
_____ ______ Visual description of how key tasks or operations are performed in space
_____ ______ Cognitive processes or decision logic typically used by organization commander or staff
_____ _______ List and description of key features of factors that impact task or mission performance
_____ ______ List of likely areas of interest where organization is likely to perform its operation
_____ ______ Description of key social characteristics of area of interest
_____ ______ Description of key physical characteristics of areas of interest
_____ ______ Terrain data for area of interest
_____ _______ List of platforms or system in each organization
_____ _______ Performance capabilities of each platform/system

_____ _______ Performance capabilities of platform subsystems
_____ _______ Visual representation of each platform

$$$ In SIM
(Y/N)

_____ _______ Other (Please specify____)
_____ _______ Other (Please specify____)
_____ _______ Other (Please specify____)
_____ _______ Other (Please specify____)
_____ _______ Other (Please specify____)
_____ _______ Other (Please specify____)
_____ _______ Other (Please specify____)

Part IIC-Desired Functional Capabilities

CMMS is being developed concurrently with most of the M&S programs that require the capability. In an ideal world, CMMS would already be available for use by your program. Acknowledging CMMS as a work in-progress: What kinds of capabilities should CMMS development provide to support future M&S programs? Of the capabilities request for hypothetical future M&S programs, which capabilities would be the highest priority candidates for concurrent CMMS development and delivery to provide concurrent support for your specific program. What technology/capability compromises would be acceptable to get an early product in-hand for your use?

Part III Architecture Requirements Survey

PART IIIA-Current Program Architecture

1) Please describe or provide documentation to describe your the current architecture you use to capture warfighting knowledge in terms of :

a) Logical structure or major software functions

b) Specific Software Products Used In Development or run-time Software Environment

c) Hardware

d) Decision Factors---What were the major factors involved in selecting this particular architecture ?

2) Do you have a logical representation of your data base ? If so, can we get it ?

3) What kind of DBMS do you use to store data ?

4) In rough terms, what is the current an expected size of your data base ? Can you break out this size estimate by type of data ?

5) Who can get access to your data ? How many people do you expect to access your data ? How frequently do you expect each user to access the data ?

6) What mechanisms do you provide to allow end-users to access data ? What were the factors that led you to this approach ?

7) Have you done studies or trades on sizing/bandwidth ? If so, can we get them ?

8) What external systems will you interface with ?

9) What standards or constraints guided the development of your architecture ?

10) Do you have performance requirements or goals for: (a) response time, (b) maintainability, (c) data quality ?. If so what are they ?

11) What architectural features do you use to facilitate security ?

12) What architectural features do you use for configuration management ?

13) What architectural lessons learned would you like to share with us ?

PART IIIB-CMMS Architecture Preferences

1) Users will access CMMS via the MSSR. Are you familiar with the MSSR ? Have you downloaded data from the MSSR ?

2) Will the simulation developers on your program have access to the Internet ? How ?

3) What kinds of hardware/software will be used to access the INTERNET ?

For the next 3 questions, assume that CMMS was populated with the simulation data from existing programs and you were tasked with building a new simulation using the warfighting knowledge available in CMMS.

4) Would you prefer to : (a) query the CMMS database and review results on line, (b) query the data base and download results in a general data format (e.g., SQL), (c) query the data base download results, and a view the results via a GUI tool ? _______.

Could you describe the factors which lead to this preference ?

5) Would you prefer to: (a) view data that has been converted to a common semantic and syntactical framework or (b) view "raw" data in same format used by the original simulation program which produced the data ? _______.

Could you describe the factors which lead to this preference ?

6) Do you have general preference for (a) view data displayed in graphical user interface readily understandable to military personnel, (b) view data in software engineering or CASE tool

format, or (c) view data in the original data base format ? _______. Under what circumstances would you prefer (a), prefer (b), prefer (c)

Could you describe the factors which lead to this preference ?

7) Suppose you were tasked to build a new simulation and as part of this effort, you were going to simulate something that was already described in CMMS and for which code had already been developed and tested ? What kinds of information would CMMS to provide for this situation:

a) Software code
b) CASE or software engineering documentation associated with code
c) Description of ‘real world” or warfighting knowledge which code was designed to represent
d) a and b
e) a, b, and c

Could you describe the factors which lead to this preference ?

8) Suppose you were tasked to build a new simulation, and as part of this effort, you were going to simulate something that was already described in CMMS and for which code had already been developed and tested. However, in this case, some modifications to the existing were software were required to reflect changes in doctrine. What kinds of information would CMMS to provide for this situation:

a) Software Code
b) CASE or software engineering documentation associated with code
c) Description of ‘real world” or warfighting knowledge which code was designed to represent
d) a and b
e) a, b, and c

Could you describe the factors which lead to this preference ?

ATTACHMENT C: JOINT INTERDICTION THREAD

This attachment describes the Joint Interdiction operation thread which will be used to demonstrate Cycle 1 CMMS capabilities. The attachment is divided into two sections. Section C.1 describes a hypothetical scenario that will be used to guide the development of the thread. Section C.2 describes the thread itself.

C.1 Hypothetical Scenario

A Joint Interdiction operation in Korea in 1998 was selected as the basis for the hypothetical scenario. Prior to selecting Korea in 1998 as a representative scenario, we examined a number of possible scenarios, including Kuwait (an expression of MRC–E), and Vietnam (an expression of an LRC Long). Korea was selected in that it represented a current emerging crisis and one which included existing command and control arrangements. This allowed a firm representation of command relationships. The year 1998 was selected in that by placing it forward in time, assumptions could be made without tripping over any real world or recent special knowledge, which could distract readers form the larger issues.

C.1.1 SITUATION OVERVIEW

Command Relationships

Command of forces in South Korea is complicated by the concurrent existence of national, combined and United Nations changes of command. Both US and ROK forces are responsible to their own governments. However, this responsibility is executed through the Commander of Combined Forces Command (CINCCFC), which provides overall theater strategic military direction. At the same time, the 1950 United Nations mandate has not expired or been revoked and the US Commander in Korea is the United Nations Commander in Chief (CINCUNC). His authority comes directly from the President of the United States, who is an agent of the United Nations.

With regard to US command relationships, in peacetime the Commander of US Forces Korea, (COMUSK) is one of several sub-unified commanders, under USCINCPAC, who in turn is responsible to the National Command Authorities (NCA). The peacetime location of COMUSK is Yongsan, Seoul, South Korea and the peacetime location of USCINCPAC is Camp H M Smith, Hawaii.

Operational and logistics support to any conflict on the Korean Peninsula would be expected to come from Commander Seventh Fleet and Commander US Forces Japan (COMUSJ). Commander Seventh Fleet (COMSEVENTHFLT), with afloat headquarters on the USS BLUE RIDGE, would conduct naval operations in support of CINCCFC. With regard to forces in Japan, they would either deploy forward into Korea, for example, F-15s from Kadena Air Base, Okinawa, or, with approval of the Government of

Japan, operate from US bases on Japan. In addition, US logistics facilities in Japan, to include repair facilities, would, again with the permission of the Government of Japan, sustain US combat operations in Korea.

Forces Available

Forces available to friendly (UN) forces included—

- The ROK ARMY
- The ROK AIR FORCE
- The ROK Navy
- The US Eighth Army (building to four divisions over 45 days). Forces external to Korea include the IXth Corps, Camp Zama, Japan, in a support role, I Corps at Ft Lewis, Washington, and the 25th Infantry Division, Schofield Barracks, Hawaii and the 6th Light Infantry Brigade, Fort Richardson, Alaska.
- The US Seventh Air Force (building to 15 fighter squadrons and three bomber squadrons over 15 days). External sources of support include Eleventh Air Force, Elmendorf AFB, Alaska and Fifth Air Force, Yokota Air Base, Japan. Each have two wings of fighter aircraft.
- The US Seventh Fleet (building from one to three CVBGs over three weeks). In addition, the ships of US Third Fleet, on the US West Coast, would be made available as required. Further, two complete Maritime Prepositioning Squadrons (MPS) would be available from their locations at Diego Garcia and Guam.
- The US III MEF, Camp Courtney, Okinawa (Not a full three brigade MEF). External sources of support include I MEF, Camp Pendleton, California.
- Special Operations forces external include 357th SOW, Kadena Air Base, Okinawa and 4th SOSC, in Washington State. Also, the US Navy has Seal Teams 1, 3 and 5 home based at Coronado, San Diego, California.
- Various allied units of moderate size.

Political and Economic Climate

The Korean Peninsula has been in tension from before the death of the Great Leader and the continuing problems with harvests, from either drought or flooding, only add to the already bleak economic picture. The continuing economic problems of Russia prevent any resumption of the previous economic support, to include a resumption of deliveries of crude oil and distillate products. The various economic trends point to a coming economic crisis in North Korea. This, in turn, threatens the political stability of the nation.

Tensions in the Western Pacific have been heightened by a number of problems involving the Peoples Republic of China, a key player in the area and a potential stabilizing influence on North Korea. Problems with the United States exist over economic considerations, including the still lingering problem of copyright protection for US intellectual properties. Differences between Taiwan and China over the place of Taiwan and the appearance of efforts toward independence have resulted in continuing

Chinese pressure on Taiwan, including the use of military demonstrations and shows of force. All efforts to resolve this problem are hampered by ongoing problems with the upcoming transition of Hong Kong to China. All of this has led to a reduction in levels of diplomatic contacts.

CRISIS DEVELOPMENT

In July of 1998 a second ship delivering US grain to North Korea disappeared under mysterious circumstances. The loss of grain put North Korea in a crisis situation with regard to food. Further, the general level of tensions in the region gave rise to a number of accusations as to who was responsible for the loss of the grain ship. Those accused included China, Taiwan, South Korea and North Korea.

By 15 August tensions had increased to the point that both the Philippines and Taiwan had undertaken mobilization. Given the limitations of their armed forces, this was purely a defensive reaction. Even so, China undertook limited mobilization in retaliation and North Korea took advantage of the cover of the crisis to conduct a major mobilization of its own.

On 18 August the US National Intelligence Officer (NIO) for Warning sent to the President a warning of war, stating that North Korea could execute a limited attack within 48 hours (50% probability) and a major attack within five days (45% probability).

In response to the developing crisis, the defense condition for Western Pacific was increased.

On 20 August the NIO for Warning upgraded his warning of war, stating that there was a 70% probability of a North Korean attack within 48 hours and an 85% probability of an attack within five days.

On 23 August North Korea attacked South Korea.

C.1.2 MISSION ANALYSIS

HIGHER COMMAND PURPOSE: This is the Joint Force Commander's understanding of the goal of his superior. It forms the basis for subsequent analysis. The selected Higher Command Purpose is:

Stop DPRK, force forces back and destroy military capabilities

NATIONAL STRATEGIC AND NATIONAL MILITARY STRATEGIC DIRECTION: A Higher Command Purpose is not the totality of national strategic direction. Directions are to:

- Maintain a Coalition (Planning Templates and Execution Templates must include allied and coalition forces.)
- Avoid Nucs, Chem, Bio (Templates for attacking enemy nuclear, chemical and biological weapons and weapons facilities would receive early consideration. Templates which included the use of friendly nuclear, chemical and biological weapons would have severe constraints on their employment (e.g., causal templates).)
- Keep confined to the Peninsula (Templates which examined horizontal escalation would be rejected.)
- Terminate on conditions favorable to ROK and US (Causal templates would have to include these considerations.)
- Avoid Long War (Risk management would be adjusted to accept more risk early on, if it allowed an opportunity to terminate the conflict early on terms favorable.)

The review in Yongsan and at Camp Smith also included a reexamination of the desired End State, in the event of war—

END STATE

After reflection, CINCCFC restated that the conflict termination requirements in Korea were:

- Status Quo Ante Bellum (restoration of pre-war boundaries).
- North Korean weapons of mass destruction, including Missiles, Nucs and Chems destroyed.

Moving from the General requirements to the specific, CINCCFC laid down those end state conditions which would allow him to easily maintain or enforce any peace agreement. These conditions were—

- Air and Maritime supremacy has been secured.
- A More Defensible Armistice Line (minimum of pre-war borders restored).
- Organized North Korean resistance has ceased.
- Both CINCCFC and the DCI agree rockets, nucs and chems destroyed (10 years to restore).
- Both CINCCFC and the DCI agree North Korean ground forces need 5 years to reconstitute.
- All North Korean submarines are accounted for, either through being sunk, destroyed, captured or interned.
- Both CINCCFC and the DCI agree North Korean SOF transport is 90% destroyed and activity in South Korea has ceased.

TASKS

Based upon the End State, a listing of specified and implied tasks was needed—

Specified Tasks

- Stop Aggression
- Restore Border
- Major Defeat of DPRK Ground Forces

Implied Tasks

- Preserve ROK Army
- Defend Seoul
- Keep Japan providing bases and over flight
- Stay on Peninsula with a POD
- Avoid nuc/chem damage requiring retaliation in kind
- Keep coalition
- Do NEO
- Preserve LOCs in ROK
- Preserve LOCs en route

C.1.3 COMMANDER'S ESTIMATE OF THE SITUATION

RESTATED MISSION

CFC/UNC forces defend ROK territory, airspace and surrounding waters, exploit opportunities for offensive action to stop aggression, restore the borders and inflict a major defeat on DPRK forces.

SITUATION

GENERAL

No Warning
No Plan
DPRK acting alone
Japan providing bases
No ongoing MRC/LRC

GEO-STRATEGIC

- Winter of 2000
- Poor US relations with Japan due to trade difficulties, however, basing agreements still exist.
- Continued instability in the Western Pacific, including continued PRC discontent with ROC (Taiwan) and with Vietnam.
- Oil discoveries confirmed in South China Sea.

ENEMY

- Assessment of window of vulnerability due to drawdown of US ground forces (down to one Bde) without concurrent buildup of US air or naval forces or compensating ROK improvements.
- No Allies
- No External Logistics Support (use stored supplies initially, then ROK assets. Serious POL limitations.)
- WMD

Nuc	Yes	5-8 Weapons
Chem	Yes	No Dong/Long Dong
Bio	No	

- Mobilization
  - ⇒ Two Weeks Covert
  - ⇒ Two Days Overt
- Broad Military Plan
  - ⇒ Chem Ports and Airfields
  - ⇒ Interdict SLOCs
  - ⇒ Force USAF off peninsula
  - ⇒ Bypass Seoul
  - ⇒ Major Ground Ops Chorwon Corridor (West Coast)
  - ⇒ Support Ground Ops Kesong and Corridors (Center and East Coast)
  - ⇒ SOF against C4I targets, leadership, airfields and ports and HARTS targeting
  - ⇒ Carrot to Japan to remain neutral with
- Broad Political Plan
  - ⇒ Fracture the coalition by attacking US support at home. Concentrate on major US casualties.
  - ⇒ Seek PRC support, offering a chance to settle old scores.
  - ⇒ Keep Japan neutral by carrot, with a backup stick of chem or nuc.
  - ⇒ Destroy basis for continuing democratic government (including destroy existing leadership).

Friendly

ASSUMPTIONS

- Japan continues to provide bases
- Allies other than US/ROK do not provide significant forces
- UN Mandate continues
- US mobilization 200K authorized (Partial Mob available?)
- Nuclear Policy
  - ⇒ Nuc=Chem=Bio
  - ⇒ No first use/No automatic response

CENTERS OF GRAVITY

Friendly

For both the United States and for South Korea the time-distance equation is a center of gravity. Needed reinforcements from the United States must travel great distances to arrive. Further, the logistic sustainment needed by the South Korean military must also come from the United States. If this equation can be lengthened, such as through submarine warfare, friendly forces face defeat. See also Reinforcement/Sustainment, below.

For Friendly forces a technological superiority is being counted upon to replace numbers of soldier on the ground and to make up for the ability of North Korea to conduct a rapid mobilization and for the large time-distance equation faced by friendly forces. Of particular importance is control of the air. This, in turn, makes airpower, and in particular, the US Air Force a center of gravity.

- ROK Army
- Reinforcement/Sustainment
  - ⇒ Air (USAF/USN/PATRIOT/ATACMS)
  - ⇒ C4I
  - ⇒ Preferred Munitions
  - ⇒ Logistics Support (including ROK Army)
- Will of the People
  - ⇒ US—Casualties
  - ⇒ ROK—?

Enemy

- C4I
- Leadership
- Ground Forces
- Logistics for Ground Forces
- Nuclear and Chem Capability

Neutrals

- Decision Making Apparatus
- Will of the People

COURSES OF ACTION

Development of Courses of Action require an understanding of the possible enemy courses of action and the development of one's own courses of action. Part of this is understanding the Enemy and Friendly Centers of Gravity.

North Korean COA: Three Courses of Action were reasonably available to North Korean forces and the third option was chosen as most likely to gain for them their political objectives.

GENERAL

- Deep Attack
- Narrow Front (West Corridor)
- Hold Center and East Corridors
- Bypass Seoul
- Topple ROK Government
- Re-unite Peninsula Under DPRK

AIR

- Limited Air Superiority
- Deliver SOF
- Attack Airbases and Ports

NAVAL

- Deliver SOF
- Conduct Submarine Campaign Against SLOCs
- Conduct Limited Landings on ROK Shore

GROUND

- Deep Attack Across Narrow Front (West Corridor)
- Limited Attack Across Center and East
- Bypass Seoul
- Defeat ROK/US Ground Forces

SOF

- Conduct Deep Operations

NUC/CHEM

- Chem on Airfields and Ports
- May Threaten Japan

(South Korean) CFC Courses of Action

In doing his Commander's Estimate of the Situation, CINCCFC examined three courses of action and compared them with possible North Korean courses of action. In doing this, he considered the impact of neutrals, such as China and Japan.

The result of his comparison of courses of action resulted in the selection of Course of Action Three, which included a defense of Seoul and the Central and East Corridors and allowed a limited penetration on the West Corridor, with a counter attack on the ground. This Course of Action was designed to allow time for initial reinforcements, so as not to require the unnecessary sacrifice of friendly forces in an attempt to hold the line in all locations. ***For its ultimate success, it depends upon the conduct of joint interdiction to destroy, delay, disrupt or divert North Korean ground combat and support forces. This would disrupt their time table and bleed their forces to the point that they could not only not continue a successful offensive operation, but would be unable to resist a counter attack by the forces of South Korea and the United States and other allied nations.***

In looking at the Course of Action, it is useful to examine the contributions of the various Service Components.

Air—In the air Course of Action Three calls for immediate actions to gain and then maintain air superiority. Air superiority is defined in this COA in the conventional manner—

> That degree of dominance in the air battle of one force over another which permits the conduct of operations by the former and its related land, sea and air forces at a given time and place without prohibitive interference by the opposing force.

Gaining and maintaining air superiority is an ongoing task and is calibrated to allow as many assets as possible to engage in other actions. Given the maritime situation, support of maritime operations is not anticipated. Close Air Support will place a moderate requirement on air assets, at least initially. Aside from certain facilities storing or producing weapons on mass destruction, North Korea has few targets of strategic value. The main effort, after the fight for air superiority will be interdiction. This will be reflected in the Air Apportionment, which will call for a heavy interdiction effort.

Maritime—For Maritime operations, Course of Action Three calls for initial efforts to ensure freedom of action, including interdiction of North Korean SOF assets moving by sea and destruction of the North Korean submarine effort. In the anti-submarine effort freedom of action is the initial requirement, with destruction of the force and its supporting facilities being a longer range goal. Naval forces will take actions to prepare for amphibious operations. This includes preparing for counter-mine operations. Naval aviation will support the hunt for North Korean SOF and submarine forces and provide protection to the fleet. Excess assets will be made available to the JFACC for tasking in accordance with CINCCFC's air apportionment.

Ground—

- Defend Seoul
- Defend in sectors in Central and East Corridors
- Allow Penetration in West Corridor to Tae Jon
- Counter Attack With Amphibious Support

SOF—

- Conduct Deep Operations
- Some Interdiction in country

C.2 JOINT INTERDICTION THREAD

THE CAMPAIGN

The overall campaign includes a major effort to gain and maintain air superiority and to prevent infiltration by North Korean SOF. A strong defense of Seoul would be made, as well as defense of the central and eastern attack corridors. The western attack corridor would be defended with a very elastic defense. A major ***joint interdiction*** effort must be mounted to destroy North Korean forces waiting to move forward and any supporting and sustaining forces.

THEATER JOINT INTERDICTION

In setting down his Campaign Plan CINCCFC will call for a major ***interdiction*** operation. This will, in turn, include interdiction efforts by air, naval, ground and SOF forces. This will be supported by space forces, both in terms of surveillance and reconnaissance, but also in terms of real time targeting.

C.2.1 AIR INTERDICTION

The bulk of theater joint interdiction will be provided by the air forces, both Air Force and Navy, with some support from Marine air assets. The thread will involve the **Execution** portion of the Air Interdiction mission assigned to the Air Force component. It will continue down to the tasks of individual aircraft executing the interdiction (e.g., delivering ordnance). More specifically, the thread will include the following:

Planning, preparation, and execution: The pilot/crew receives mission planning data from the MPC and prepares for the mission--standard procedures (taxi, takeoff computations, weather reports, notices, etc.) and mission specific procedures (tactical route study, target study, weapon delivery, etc.). Meanwhile, the aircraft is being prepared by maintenance--standard prep (fuel and other servicing) and mission specific prep (weapon prep and loading, etc.). The pilot/crew then arrive at the aircraft, taxi, arm, and takeoff; proceed to the target area via a preplanned route selection (which, in this text case includes an air refueling task); in the target area, execute the mission attack profile (which, in this text view is a "pop up" attack); execute safe escape maneuver after weapon release; egress the target area; proceed via preplanned routing to return to base; execute procedural approach to a tactical recovery and routine landing; dearm the aircraft; park the aircraft and shut down engines; mission complete; proceed to debriefings.

C.2.2 Land Interdiction

Land Interdiction is implemented via a field artillery attack by an ATACMS battalion.

The corps artillery headquarters receives a tasking from ARFOR to conduct an attack against a fixed target (command post bunker complex) 75 kilometers north of the FLOT. The time on target and desired destruction criteria have been specified. Three structures within a football sized area are the targets. They contain a mix of masonry and wooden structures with a few windows. The complex also includes communications antennas for Microwave radio. Terrain is Piedmont: rolling hills, sparsely vegetated.

The Corps artillery HQ passes the target to the ATACMS battalion. Mission planning begins in the battalion FDIC. Meteorological data is obtained, missile flight plan (including time on target, probable damage) is computed, and planning is completed 20 minutes prior to required time on target. Time of flight is computed and an offset launch time is determined.

ATACMS Battalion approves the mission data and it is transferred to the selected firing battery for entry into the missile control system. A primary and backup missile is readied. Both systems load data successfully and are ready for final launch sequence on command.

At one minute prior to launch, final checks are made; the primary missile is launched as scheduled; backup is kept in readiness for use as potential restrike if required.

Overhead assets determine 40% target destruction. The JTCB determines that a restrike is required. This is passed down through channels to the firing launcher where the backup systems is launched. BDA is satisfactory and the mission is ended. The launchers relocate to their alternate positions to avoid an enemy strike from target acquisition. They will reload, depending on availability of missiles at the site.

C.2.3 Maritime Interdiction

Maritime Interdiction is implemented via a TLAM attack by a surface ship. The Commander, Task Group Kearley, receives a tasking from the Naval Component Commander to conduct a TLAM strike against radar installations 135 miles inland from the coast and 75 miles north of the FLOT. Mission data, to include time on target and destruct criteria, are specified. The target area is heavily defended with SAM and tactical air. The target is a key link in the air defenses which must be destroyed prior to upcoming air operations. The TLAM was selected by the Joint Target Coordinating Board (JTCB) due to time on target and vulnerability of manned systems considerations. The TLAM strike is part of a coordinated (land, air and naval) joint interdiction operation and is time critical.

As the order is received at the NAVFOR command ship, the Afloat Commander and staff consult satellite photo coverage, confirm target location; work up the metro and inland air defenses along stored mission route; consider best route with least defenses; review ROE (not a problem here); and determine that the executing ship has an acceptable mission and does not need a mission update.

Task Group Kearley is directed to execute. TG Kearley immediately begins development of an engagement plan, and initiates movement to the launch basket and positions to reach first preplanned (landfall) waypoint. TG maintains Battle Space Dominance. The Aegis cruiser commences the TLAM launch sequence and loads the mission data on to four missiles (three primary and a backup). The launch sequence is held at launch minus 10.

The firing order is given at the designated time. The missiles all function correctly and are tracked in flight to the target by E3 aircraft and other assets. The TG stands by for possible restrike. Overhead BDA reports satisfactory target destruction. The missile launch ship stands down the spare missile, reports mission complete, and steams to a new location with the Task Group.